HP X.25/9000 User's Guide

HP-UX 11i v3

Edition 8

Manufacturing Part Number: J2793-90072

February 2007

Legal Notices

Contents

Contents

Contents

Tables

Tables

Tables

Tables

Figures

Figures

1 About the X.25 Product

Introduction

The Hewlett-Packard X.25 link for HP 9000 systems provides networking link hardware and software to allow HP computer systems to communicate with other HP and non-HP computers over X.25 packet switching networks.

The X.25 link implements the *CCITT X.25 Recommendations* and contains the components necessary to connect an HP 9000 to a public or private packet switching network conforming to the *CCITT X.25 Recommendation (1980, 1984 or 1988*), or to another system in a back-to-back configuration.

The information in this manual applies to HP 9000 Series 700 and 800 systems that use single, dual, or quad-port communications hardware. Any differences are specifically noted.

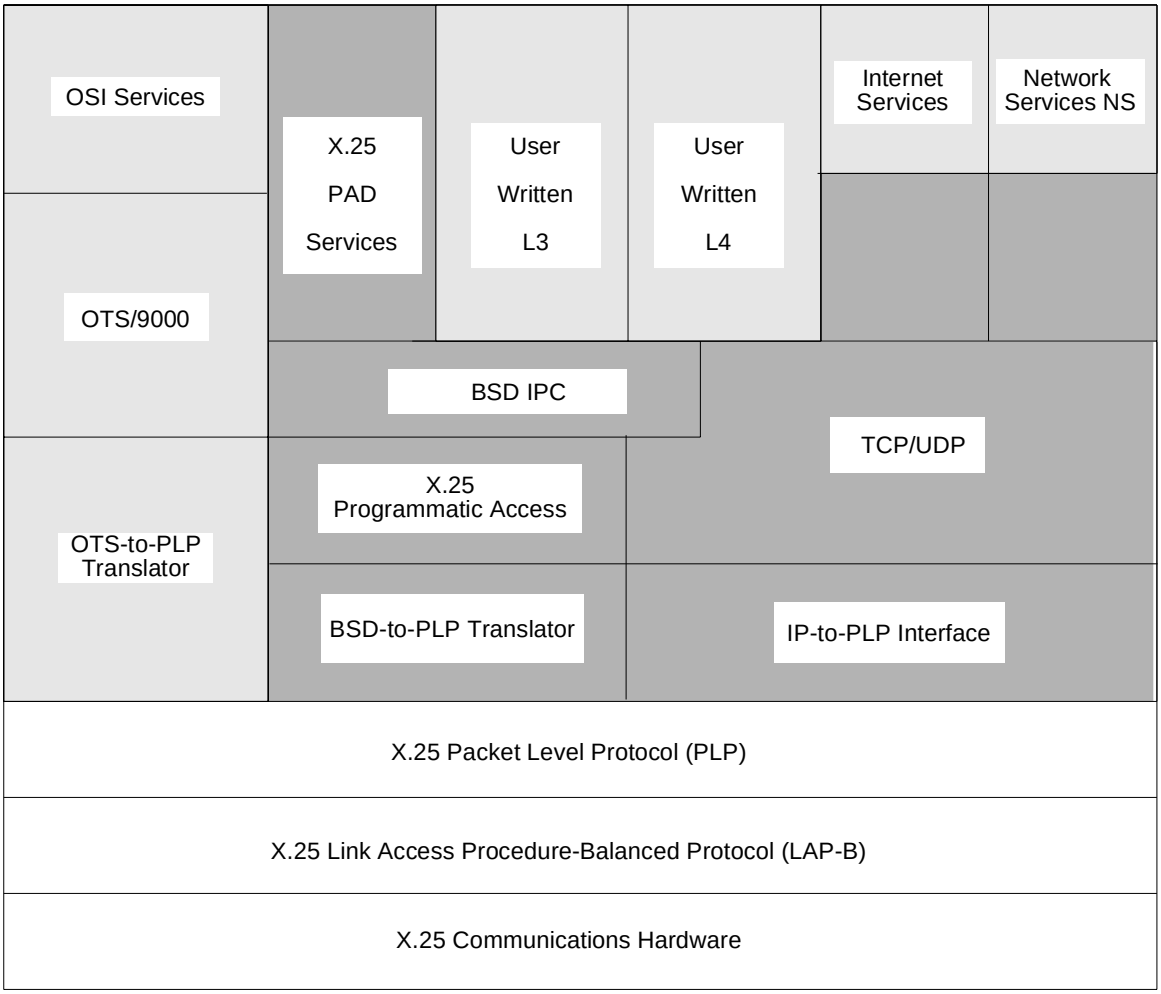
NOTE In a diskless cluster, X.25 software is only supported on the server system. It is not supported on client systems for this type of environment.

The following diagram illustrates the X.25 link architecture where:

- solid white boxes represent the product's integral components.
- lightly shaded boxes represent the product's protocol components.
- heavily shaded boxes represent optional components (external to the product) that can be added to the product.

Each component depends on the component(s) below it and must be installed and running in order to support higher-level components.

Figure 1-1 X.25 Link Architecture



Application (L7), Presentation (L6), and Session (L5) Levels

The X.25 link does not provide any components for the general support of the application and presentation levels (levels 7 and 6, respectively), although X.25/9000 PAD Services do provide some of the functionality of these levels.

User-written application programs, Internet Services/Berkeley Services (via BSD Sockets) and NS (via NetIPC Sockets), are accessed by means of TCP or UDP Transport Level (level 4) protocols.

For application level services, you can install OSI Services to run over the X.25 network (for more information, refer to OSI Services documentation). The OTS/9000 product provides access to X.25 for OSI Services. You can install application level services such as Internet/Berkeley Services and Network Services (for more information, refer to the documentation for these products).

BSD IPC (Berkeley Software Distribution InterProcess Communication) allows direct programmatic access to the X.25 packet level (level 3), or TCP/UDP at the transport level (level 4), for user-written application programs.

NetIPC sockets also provides a programmatic interface to TCP/UDP at the transport level.

Transport Level (L4)

At the transport level (level 4), the X.25 link provides TCP (based on the DARPA standard) and UDP. These Transport level protocols are used by Internet/Berkeley Services, by NS, and by user application programs that access the TCP/UDP and IP protocols.

The TCP protocol is a connection-based protocol. TCP verifies that all data is delivered without duplication to its destination. The UDP protocol, unlike TCP, has no concept of a connection. Messages are sent as a unit with source and destination information in the header.

The X.25 link provides a BSD to PLP (Packet Level Protocol) translator to allow access to PLP at level 3 for user-written application programs via BSD IPC.

The X.25 link also provides an OTS to PLP translator to allow access to PLP at level 3 for OSI Services via Xport OSI.

Packet/network Level (L3)

At the packet/network level (L3), X.25 link provides direct X.25 programmatic access via BSD IPC. For full details on X.25 programmatic access, refer to the *X.25/9000 Programmer's guide* (part number: J2793-90065).

The X.25 link also provides IP access to TCP or UDP Transport protocols for programs such as Internet/Berkeley Services and NS, allowing communication over X.25 in accordance with RFC 877.

Data link (L2) and Physical (L1) Levels

At the data link level (level 2), the X.25 link provides the LAP-B (Link Access Procedure-Balanced) protocol. LAP-B is a data link protocol, specified by the 1980 CCITT X.25 recommendations, that determines frame exchange procedures.

At the physical level (level 1), the X.25 link provides support for X.21, X.21bis, and V.35 interfaces (depending on your particular communications hardware). These sets of recommendations define the standards for X.25 at the physical level and apply specifically to connections to a packet switching network. X.21bis (equivalent to V.24 and RS-232) is applicable to X.25 physical interfaces with transmission speeds up to 64 kb/s. V.35, RS-449, X.21, and RS-530 supports transmission speeds up to 2 Mb/s. Other standards may also be supported as they become available.

2 Installation

Before You Install the Software

Before installing the X.25 link software, check the requirements below to make sure that all required software and hardware has been correctly installed and configured.

NOTE This product is only supported on the system that is acting as a server. It is not supported on client systems.

Hardware Requirements

This section describes the hardware requirements of the J2793B X.25 software for HP 9000 server systems.

If you have not already done so, install the X.25 interface card as described in the hardware installation guide for your X.25 product. If you are installing multiple X.25 cards, check the requirements in "Installing Multiple X.25 Interface Cards" below.

If a modem connection is required, connect the X.25 card to the modem or the modem eliminator according either to the modem vendor's specification sheet, or the instructions provided by the network provider.

Installing Multiple X.25 Interface Cards

When installing multiple X.25 cards,

- Always install the cards in adjacent slots in ascending slot order.
- Always install X.25 cards with the system shut down.
- Do not skip slots between X.25 cards.

Hardware Compatibility

- HP 9000 PCI bus

The High Availability feature can run on:

- PCI hardware cards, Product Numbers J3525A (2-ports)

OS Platform and Version Compatibility

The version of the X.25 link software you're installing must be compatible with the version of HP-UX you're running (for example, HP-UX version 11i v3 for B.11.31.01)

- Disk space required to install: 7 Mb
- Software install with system up or down? Up
- Single-user state required or recommended? No
- Reboot? Yes

Memory

Your X.25 link supports a high number of virtual circuits operating at high baud rates. Since both X.25 and the BSD sockets API may store a certain amount of data for each socket/circuit, HP recommends that you check that your system has enough memory to handle the number of VCs you plan to use. A few guidelines to help you are provided below.

Related Parameters

- Level 3 window size (W in formulas below) as configured in the `x25config` file
- Level 3 packet size (P in formulas below) as configured in X.25 configuration file
- Socket buffer size (B in formulas below) used in your applications (`setsockopt()` system call). The default is 4 Kb.

Evaluation Formulas The following formulas can be used to evaluate X.25 memory requirements:

Memory for each VC (MVC) in bytes:

MVC = (B x 3) + (2 x (W + 1) x P)

Total Memory (TM) for X.25 in bytes:

TM = sum (MVC) + Number of cards x 5120

NOTE These figures are only for protocol and API requirements. You should also consider the memory required by your applications.

Shortcut Method Assuming B = P for all VCs, TM can be rounded to:

- If B < 512, TM = approx. (1,536 + (2 x W + 2) x B) x No. of VCs
- If B > or = 512, TM = approx. (2 x W + 5) x B x No. of VCs

The following table (in bytes/VC) provides a quick guide:

Table 2-1

		Buffer Size (B) = Packet Size (P) in bytes					
		128	256	512	1024	2048	4096
Window (W) size	1	2 K	2.5 K	3.5 K	7 K	14 K	28 K
	2	2.3 K	3 K	4.5 K	9 K	18 K	36 K
	3	2.5 K	3.5 K	5.5 K	11 K	22 K	44 K
	4	2.8 K	4 K	6.5 K	13 K	26 K	52 K
	5	3 K	4.5 K	7.5 K	15 K	30 K	60 K
	6	3.3 K	5 K	8.5 K	17 K	34 K	68 K
	7	3.5 K	5.5 K	9.5 K	19 K	38 K	76 K

Example 2-1 **Example:**

A system has two X.25 cards configured as follows:

Card 1:

Window = 2
Packet size = 1024
Socket buffer size = 4096 (default)
200 VCs are used

Card 2:

Window = 4
Packet size = 128
Socket buffer size = 4096 (default)
400 VCs are used

TM = (4096 x 3) + (2 x 3 x 1024)) x 200
+ ((4096 x 3) + (2 x 5 x 128)) x 400
+ (2 x 5120)

TM = 9,123,840 (8.7 Mb)

Software Requirements

Before installing the X.25 link product, make sure that the software listed below has been correctly installed on your system. Refer to the related publication if you need more information about any of these products. If you cannot find the software or information you need, contact your HP representative.

- HP-UX operating system version 11i v3 – see *Installation and Update for HP Integrity Servers and HP 9000 servers*.
- Internet Services – see *HP-UX Internet Services Administrator's Guide: HP-UX 11i v3*

Installing the X.25 Link Software

Follow the steps below to install the X.25 link software:

Step 1. Insert the software media (tape or disk) in the appropriate drive.

Step 2. Type: **`swinstall`**.

(See the man page on `swinstall` for more information on this command).

Step 3. Click on `OK` on the "Specify Source" window.

Step 4. Highlight `J2793B` in the "Software Selection" dialog, then select `Mark For Install` from the `Actions` menu to install all file sets in the bundle.

If you want to select only certain file sets, double-click on the product name to access the file sets that you want to mark for installation (each time you double-click, you go down one level in the bundle structure).

Step 5. When you have marked the product components you want to install, select `Install (analysis)` from the "Actions" menu.

Step 6. When you have successfully completed the analysis, click on `OK` from the `Analysis` dialog to load the X.25 file sets.

The `swinstall` utility loads the file sets, runs the customized scripts for the file set, and builds the kernel. Estimated time for processing: 8 to 10 minutes.

If the kernel build is not successful, the `swinstall` program returns you to a new shell. The cause of the failure will appear at the end of the `/var/adm/sw/swinstall.log` file.

3 Configuration

Configuring the X.25 Link

This section describes how to configure your X.25 link using HP System Management Homepage(SMH).

HP SMH provides Graphical User Interface (GUI), Terminal User Interface (TUI) and Command Line Interface (CLI) for managing HP-UX. You can access these interfaces using the `smh` command (/usr/sbin/smh).

If the DISPLAY environment variable is set, HP SMH opens in the default web browser. If the DISPLAY environment variable is not set, HP SMH opens in the TUI. For more information, see the HP-UX 11i v3 release notes and the SMH product online help.

In SMH, the term “card” refers to a particular X.25 interface. Dual-port cards have two interfaces. In this context, an interface is the same as a port.

NOTE System Administration Manager (SAM) is deprecated on HP-UX 11i v3 and replaced with the enhanced HP System Management Homepage (HP SMH). Users who attempt to start SAM from the command line interface are automatically redirected to the SMH user interface. SMH can be run directly in a web browser window by entering http://hostname:2301. In the Graphical User Interface (GUI), after logging in, the SMH main menu is displayed. Select Tools > networking and communications > network interface cards > choose X.25. For more information on SMH, see the HP System Management Homepage Release Notes (Part Number:381383-009) on http://docs.hp.com.

Using SMH

Follow the steps below to start SMH and display the “Configure X.25 Card” window:

Step 1. Make sure that you are logged in as root. Then, at the HP-UX prompt, enter:

```
smh
```

(To run SMH in the background, type: **`smh &`**).

Step 2. At the SMH main window, select "Tools" menu. SMH displays numerous object lists.

Step 3. Double-click on the `Network Services Configuration` menu. SMH displays an object list that shows all network interfaces (devices) installed in your system.

SMH displays each port (for multi-port cards) as a unique interface with its hardware path and name. Interfaces are listed in order of their slot number.

Step 4. Highlight the X.25 device that you want to configure on the object list and select `Configure` from the `Actions` menu, or double-click the device you want.

The "Configure X.25 Software" window displays. If you are modifying a device that is already configured, the window is entitled, "Configure X.25 Card."

NOTE Refer to Appendix A, Using Non-English Subscription Forms, for the English equivalents of the French, Italian, German, and Spanish fields that appear on non-English subscription forms.

Using SMH's On-line Help

The SMH on-line Help provides information (descriptions, formatting, and ranges) for all fields. You can access the SMH on-line Help system by:

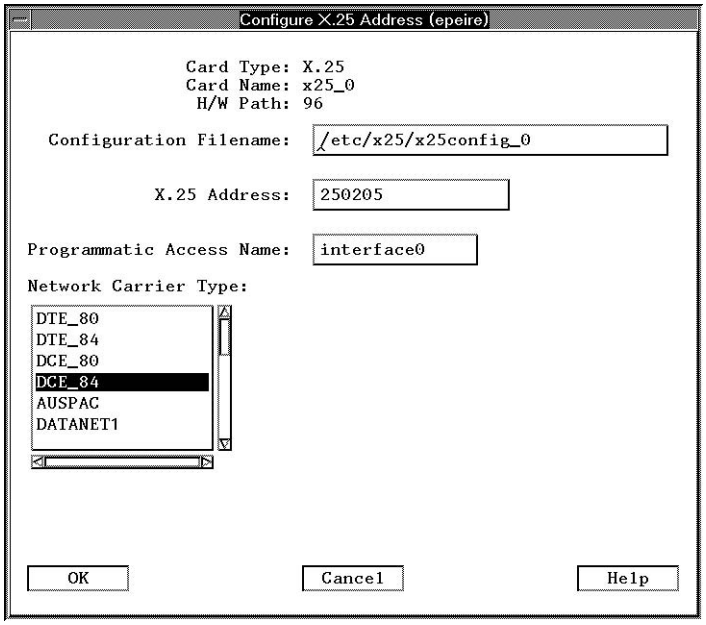
- Clicking on the `Help` button in a dialog or message box to display information about how to use the dialog, or about the message.
- Pressing **`F1`** to display information about the object selected by the cursor (for example, a data entry field).
- Selecting an item from the `Help` menu (located on the menu bar). You can display information about the current SMH dialog, keyboard navigation within SMH, using the SMH Help system, and the version of SMH you are currently running.

Configuring an X.25 Address

Follow the steps below to configure an X.25 address:

Step 1. From the "Configure X.25 Software" window, select `Configure X.25 Address`. The following dialog appears:

Figure 3-1



NOTE The SMH windows and dialogs shown on these pages are intended only as examples. The information that appears in your SMH dialogs, such as the `Card Name` and `Programmatic Access Name`, depends on your particular communications hardware.

Step 2. Enter or modify the field values as required (refer to the field descriptions below).

Table 3-1 Field Descriptions

Configuration File name	Name of the file that will contain the parameters for configuring the interface (a physical port). If you are configuring more than one interface, specify a unique configuration file for each interface. The configuration file must be named `x25config_`*n*`p`*x*, where *n* represents the communications card number (use 0 for the first card, increasing the value by 1 for each additional card up to 255), `p` is a place marker, and *x* is the port number (1 to 4). Note that `p` and *x* are only required for systems with dual-port or quad-port cards.
X.25 Address	Address assigned to each local X.25 interface (card or port) by the network carrier. X.25 checks it for diagnostic and identification purposes only. This address is sometimes referred to as the X.121 address. Use the value given on your subscription form.
Programmatic Access Name	Name given to the interface you are configuring (used for X.25 level 3 programmatic access).
Network Carrier Type	Type of network to which the X.25 interface is attached (as it appears on your subscription form). A complete list of possible networks is shown below. The default is DTE_84.
X.25 Packet Address	This field appears (with its default value) only if you select TRANSPAC as the Network Carrier Type.

Network types

You *must* select a Network Carrier Type that matches the type of network to which you are connected.

```
DCE_80      DCE_84           DCE_88           DTE_80
DTE_84      DTE_88           AUSPAC           DATANET1
DATAPAC     DATEXP_AUSTRIA   DATEXP_DEUTSCHE  DCS
DDN         DDXP             HPPPN            ITAPAC
LUXPAC      PSS              TELENET          TRANSPAC
TYMNET      IBERPAC          TELEPAC          DATAPAK
```

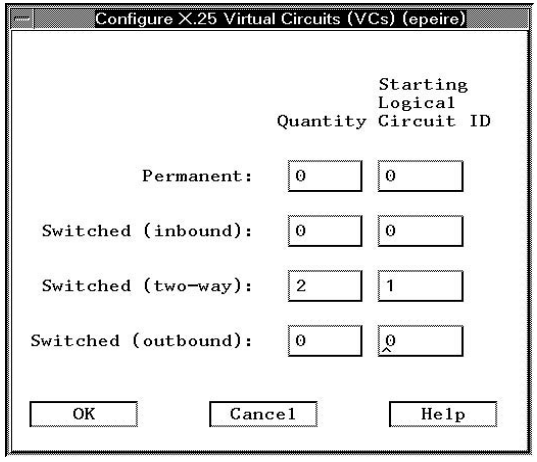
Step 3. Click on `OK` to return to the "Configure X.25 Card" window when you have finished configuring the X.25 Address.

Configuring X.25 Virtual Circuits (VCs)

Follow the steps below to configure X.25 virtual circuits:

Step 1. From the "Configure X.25 Card" window, select `Configure Virtual Circuits`. The following dialog appears:

Figure 3-2



Step 2. Enter or modify field values as required (refer to the field descriptions below).

When you configure the Quantity column (starting with the number of Permanent VCs), SMH automatically fills in the starting Logical Circuit Identification (LCI) as you **Tab** between fields.

Table 3-2 Field Descriptions

Permanent - Quantity	Number of Permanent VCs. Use the value given on your subscription form.
Switched (inbound) - Quantity	Number of Switched (inbound) VCs. Use the value given on your subscription form.
Switched (two-way) - Quantity	Number of Switched (two-way) VCs. Use the value given on your subscription form.
Switched (outbound) - Quantity	Number of Switched (outbound) VCs. Use the value given on your subscription form.

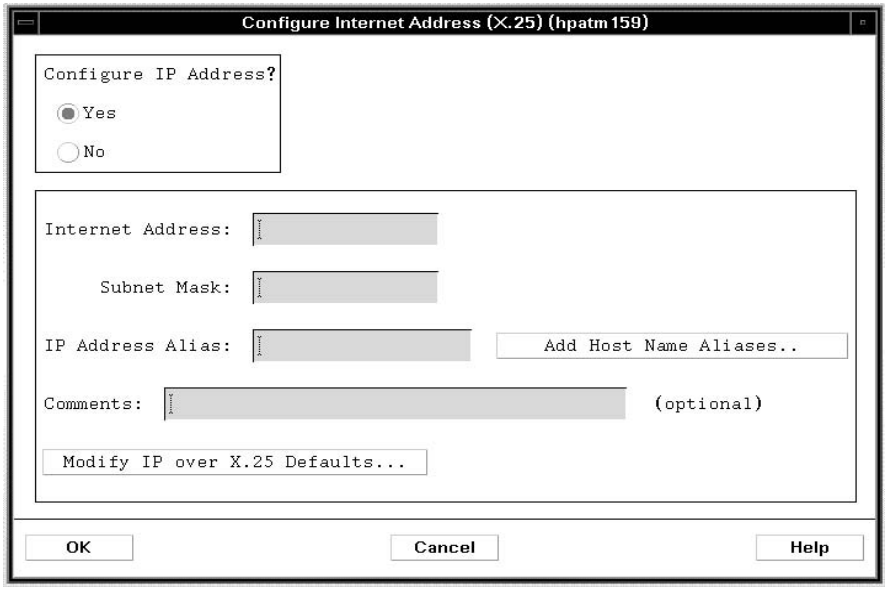
Step 3. Click on `OK` to return to the "Configure X.25 Card" window when you have finished configuring X.25 Virtual Circuits.

Configuring an Internet Address

Follow the steps below to configure an Internet address:

Step 1. From the "Configure X.25 Card" window, select `Configure Internet Address`. The following dialog appears:

Figure 3-3 Configure Internet Address



The "Configure Internet Address" dialog gives you a **yes** or **no** option to configure an IP address for this port.

Step 2. Choose `Yes` or `No` to configure an IP address.

If you select `No`, you indicate that no IP address is associated with this X.25 port. When you select `No`, all other fields in this dialog disappears.

If you select `Yes`, you must fill in the other required fields in this dialog.

Step 3. Enter or modify field values as required (refer to the following field descriptions).

Table 3-3 Configure Internet Address Field Descriptions

Internet Address	The identifier by which this interface (port) is known on the network. It consists of four sets of integer values (0 to 255) separated by periods (for example, 192.2.3.6). The IP address must be unique for each X.25 interface. It must also specify a different subnet from the IP address of any other X.25 or LAN interface on this system. To obtain an IP address, see your system administrator, network administrator, or HP representative.
Subnet Mask	The subnet mask (like IP addresses) is composed of four integers (0 to 255) separated by periods. The subnet mask is used for routing. When you enter an IP address, SMH places a default subnet mask in the subnet mask field, depending on the class of IP address you enter. You may use the default or enter another one if required. Table 3-4, "IP Address and Default Subnet Masks," shows the IP addresses listed by class and their default subnet masks).
IP Address Alias	The symbolic name (in alphanumeric format) by which this network interface will be known on the network. Use the `Add/Modify Host Name Aliases` button to add or modify aliases.

Table 3-4 IP Address and Default Subnet Masks

IP Address	Class	Default Subnet Mask
A.*.*.* (A between 1 and 127)	A	255.0.0.0

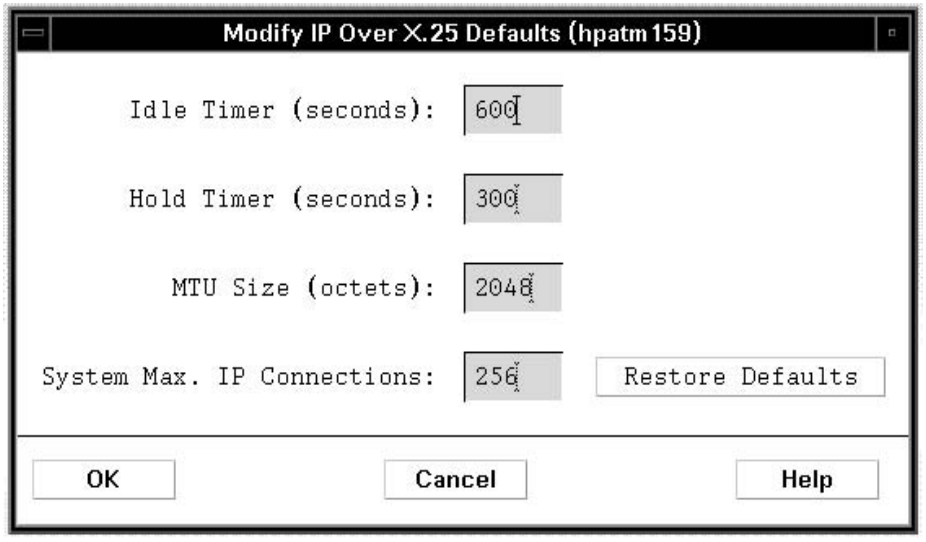
Table 3-4 IP Address and Default Subnet Masks (Continued)

IP Address	Class	Default Subnet Mask
A.B.*.* (A between 128 and 191) (B between 0 and 254)	B	255.255.0.0
A.B.C.* (A between 192 and 239) (B,C between 0 and 254)	C	255.255.255.0
A.B.C.* (A between 240 and 254) (B,C between 0 and 254)	D	Not Allowed

The Internet address is composed of two addresses: the network address and the subaddress. Zero (0) and -1 are not allowed in the subaddress.

Step 4. If required, click on `Modify IP over X.25 Defaults` to modify the `Idle Timer`, `Hold Timer`, `MTU Size` and `System Max. IP Connections` defaults.

Figure 3-4 Modify IP Over X.25 Defaults



Enter or modify field values as required (refer to the field descriptions below).

Table 3-5 Modify IP over X.25 Defaults Field Descriptions

Idle Timer	Sets the number of seconds a circuit stands idle before it is cleared by IP. The range is 0 to 32767. The default is 600.
Hold Timer	Sets the number of seconds a circuit may be inactive before it is designated as inactive. Inactive circuits may be cleared when all other circuits are unavailable and a connection request is received by IP. *Do not set the hold timer to a value greater than the idle timer*. The range is 0 to 32767. The default is 300.
MTU Size	The maximum transmission unit size in octets (bytes). The range is 20 to 8192. The default is 2048. For DDN configured interfaces, it must be less than or equal to 1007.
System Max. IP Connections	The maximum number of IP connections that can be simultaneously active for the whole system. This is a global parameter. The default value for this parameter is 256 connections.

Step 5. Click on `OK` to return to the "Configure X.25 Card" window when you have finished configuring the Internet address.

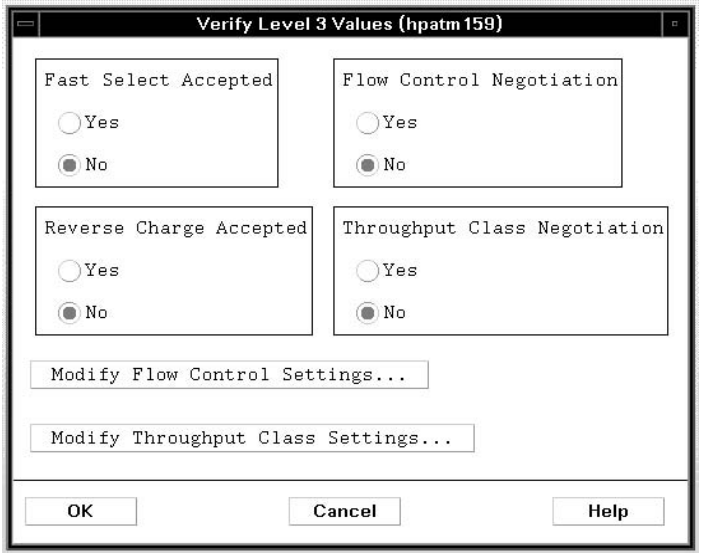
Verifying Level 3 Values

If you subscribe to a public network, the network provider will supply the appropriate settings for most level 3 parameters. These settings will differ depending on the network provider and the type of service to which you subscribe. Refer to your Network Subscription Form for the correct values for your configuration.

Follow the steps below to verify level 3 values:

Step 1. At the "Configure X.25 Card" window, select `Verify Level 3 Values`. The following dialog appears:

Figure 3-5 Verify Level 3 Values



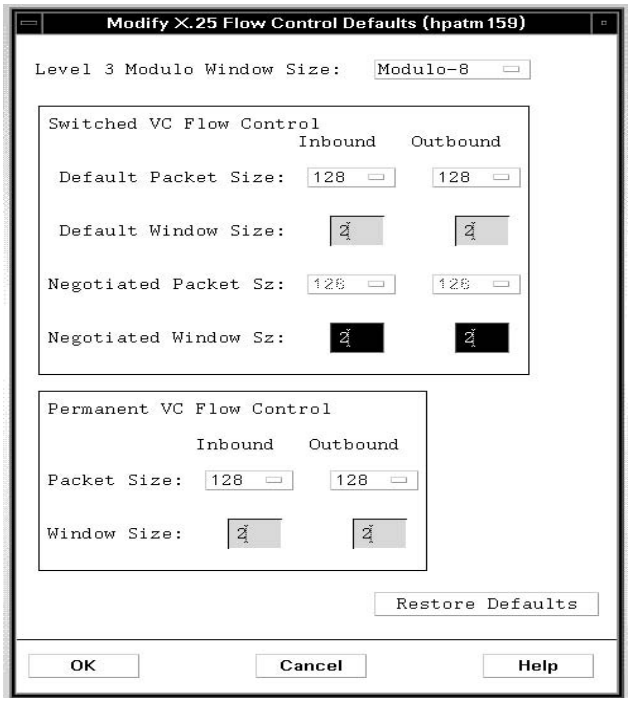
Step 2. Enter or modify field values as required (refer to the field descriptions below).

Table 3-6 Verify Level 3 Values Field Descriptions

Fast Select Accepted	If fast select is enabled (Yes), up to 128 octets of information can be transferred in call request and clear packets. The default is No (disabled). See your subscription form.
Flow Control Negotiation	The default is No (disabled). Use the value given on your subscription form.
Reverse Charge Accepted	If reverse charge is enabled (Yes), reverse charge calls can be accepted by the application. If reverse charge is disabled (No), X.25 rejects reverse charge calls automatically. The default is No. Use the value given on your subscription form.
Throughput Class Negotiation	Throughput class refers to line speed. If it is enabled (Yes), the Switched VC Negotiated value is used; if disabled (No), no throughput class negotiation is accepted. The default is No.

Step 3. Click on `Modify Flow Control Settings` to modify Switched and Permanent VC flow control settings. The following dialog appears:

Figure 3-6



Step 4. Select either Modulo-8 or Modulo-128 to set level 3 Modulo parameters. The default is Modulo-8.

The valid window-size values for Modulo-8 range from 1 to 7. For Modulo-128, the valid window-size values range from 1 to 127.

NOTE Flow Control Negotiation must be enabled on the "Verify Level 3 Values" dialog to allow the use of the negotiated flow control fields. If you do not enable Flow Control Negotiation, the Default Packet Size and Default Window Size values will be used.

Enter or modify field values as required (refer to the field descriptions below).

Table 3-7 Switched VC Flow Control Field Descriptions

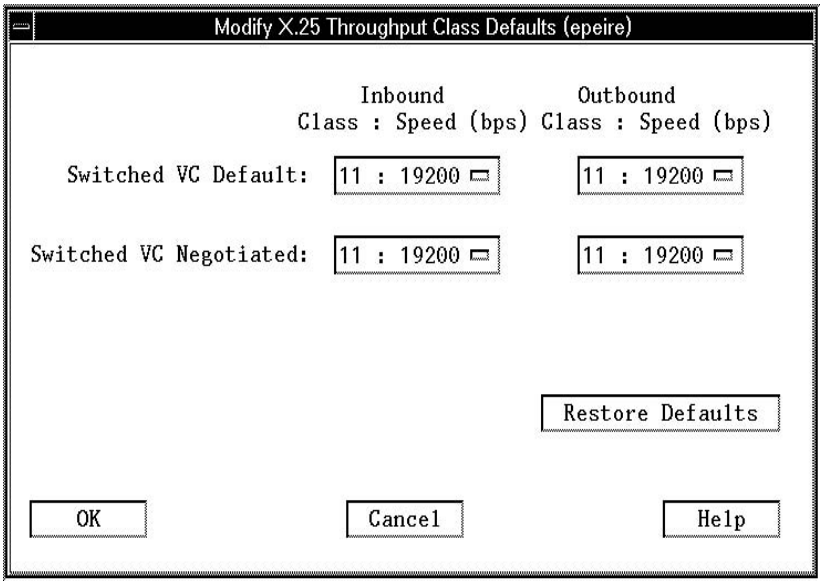
Default Packet Size: Inbound & Outbound	Maximum packet size to be used over an SVC. The range is 16 through 4096 octets. The default is 128 for inbound and outbound packets. Use the value given on your subscription form.
Default Window Size: Inbound & Outbound	Maximum number of default packets that can be transmitted without acknowledgment over an SVC. The range is 1 through 7 for Modulo-8, and 1 through 127 for Modulo-128. (We recommend a value greater than 7 for Modulo-128). The default is 2 for inbound and outbound transmission. Use the value given on your subscription form.
Negotiated Packet Size: Inbound & Outbound	Maximum packet size to be used over an SVC. The range is 16 through 4096 octets. The default is 128 for inbound and outbound packets.
Negotiated Window Size: Inbound & Outbound	Maximum number of default packets that can be transmitted without acknowledgment over an SVC. The range is 1 through 7 for Modulo-8, and 1 through 127 for Modulo-128. (We recommend a value greater than 7 for Modulo-128). The default is 2 for inbound and outbound transmission.

Table 3-8 Permanent VC Flow Control Field Descriptions

Packet Size: Inbound & Outbound	Maximum packet size to be used over a PVC. The range is 16 through 4096 octets. The default is 128 for inbound and outbound packets. Use the value given on your subscription form.
Window Size: Inbound & Outbound	Maximum number of packets that can be transmitted without acknowledgment over a PVC. The range is 1 through 7 for Modulo-8, and 1 through 127 for Modulo-128. (We recommend a value greater than 7 for Modulo-128). The default is 2 for inbound and outbound transmission.

Step 5. Click on `Modify Throughput Class Settings` to modify Switched and Permanent VC throughput class settings. The following dialog appears:

Figure 3-7



Step 6. Enter or modify field values as required (refer to the field descriptions below).

Table 3-9 Modify Throughput Class Settings Field Descriptions

Switched VC Default: Inbound and Outbound	CCITT class number. If Throughput Class negotiation is disabled, this value replaces the Switched VC Negotiated field value. The range is 3 through 13 and the default is 11. See the table after these field descriptions listing the CCITT class numbers and the corresponding line speed Baud rate. Use the value given on your subscription form.

Table 3-9 Modify Throughput Class Settings Field Descriptions

Switched VC Negotiated: Inbound and Outbound	CCITT class number. If Throughput Class negotiation is enabled, this value is used as the opening bid for outbound calls and as a counter offer when the inbound opening bid is higher. The range is 3 through 13, and the default is 11. See the table below for CCITT class numbers and corresponding line speed Baud rate. Use the value given on your subscription form.

Table 3-10 Throughput Classes and Line Speeds

CCITT class number	Line speed Baud rate (bps)
3	75
4	150
5	300
6	600
7	1200
8	2400
9	4800
10	9600
11	19200
12	48000
13	64000

Step 7. Click on `OK` to return to the "Configure X.25 Card" window when you have finished verifying level 3 values.

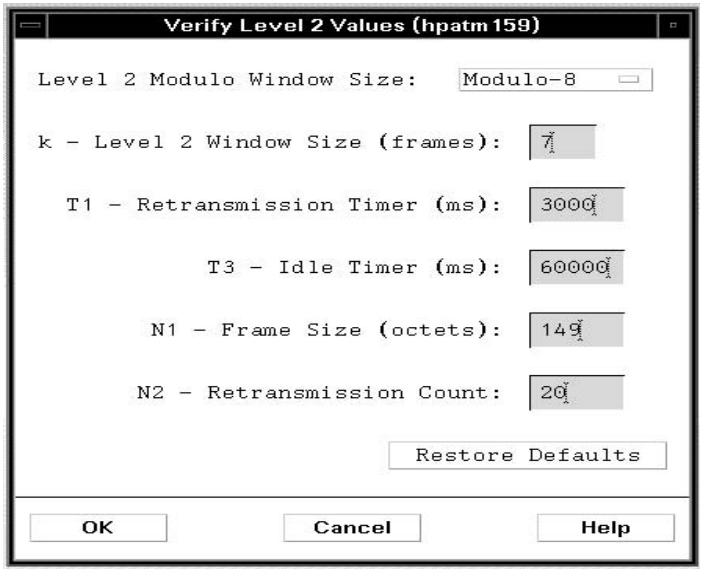
Verifying Level 2 Values

Follow the steps below to verify level 2 values:

NOTE If you subscribe to a public network, the network provider will provide the appropriate settings for all level 2 parameters. These settings will differ depending on the network provider and the type of service to which you subscribe. Refer to your Network Subscription Form for the correct settings for your configuration.

Step 1. At the "Configure X.25 Card" window, select `Verify Level 2 Values`. The following dialog appears:

Figure 3-8



Select either Modulo-8 or Modulo-128 to set the Level 2 Modulo values. The default is Modulo-8.

The valid window-size values for Modulo-8 range from 1 to 7. For Modulo-128, the valid window-size values range from 1 to 127.

Step 2. Enter or modify field values as required (refer to the field descriptions below). Use the `Help` button to display information and instructions for each field.

Table 3-11 Verify Level 2 Values Field Descriptions

k - Level 2 Window Size	Maximum number of frames that can be transmitted without an acknowledgment. The range is 1 through 7 for Modulo-8 and 1 through 127 for Modulo-128. The default is 7. Use the value given on your subscription form.
T1 - Retransmission Timer	Maximum number of milliseconds to wait for an acknowledgment before retransmitting a frame. The range is 100 to 12000. The default is 3000. Use the value given on your subscription form.
T3 - Idle Timer	Maximum number of milliseconds that a line can be idle before it is declared disconnected. This value should be greater than or equal to the Retransmission Timer (T1) times the Retransmission Count (N2). The range is 1000 to 240000. The default is 60000. Use the value given on your subscription form.
N1 - Frame Size	Maximum number of octets that can be transmitted in a single frame. The range is 149 (minimum) through 4103. The default is 149.
N2 - Retransmission Count	Maximum number of times a given frame can be transmitted before an error condition is identified. The range is 0 through 255. The default is 20.

NOTE Because the timer values count the amount of time between unacknowledged frames, you may need to increase the values of T1 (Retransmission Timer) and T3 (Idle Timer) if you use Modulo-128 with a window size greater than seven.

Step 3. Click on `OK` to return to the “Configure X.25 Card” dialog when you have finished verifying level 2 values.

Step 4. Click on `OK` to complete X.25 software configuration and save your changes. Your X.25 interface (port) should appear in the object list with status `Enabled` (no problem found and link connected correctly). If not, carefully repeat the steps in this chapter until the configuration is enabled.

If your port is not connected to a running network, it will display as `configured`.

You have completed the interface configuration.

Step 5. Select `Exit` from the `File` menu if you have no need to configure access to other systems or to PAD services. Then exit SMH.

If you need to configure access to other systems or configure PAD services, stay in SMH and continue with the instructions in "Configuring Remote System Access" (to configure access to other systems) and in "Configuring PAD Services" (to configure PAD services).

NOTE Your system will *not* reboot when you exit. Your configuration is effective immediately without the need to create a new kernel.

If you want to control Services (Internet/Berkeley Services or Network Services (NS)), refer to the documentation set for those products.

Configuring Remote System Access

Follow the steps below to configure access to other hosts and systems that use TCP/IP protocol. (The procedures in this section are optional. Your X.25 link software does not require that you configure access to other systems).

NOTE System Administration Manager (SAM) is deprecated in the 11iv3 release of HP-UX. HP System Management Homepage (HP SMH), an enhanced version of SAM, is introduced for managing HP-UX.

Step 1. If you have not already done so, type **`smh`** at the HP-UX prompt.

Step 2. At the SMH main window, select "`Tools`". Numerous object lists are displayed.

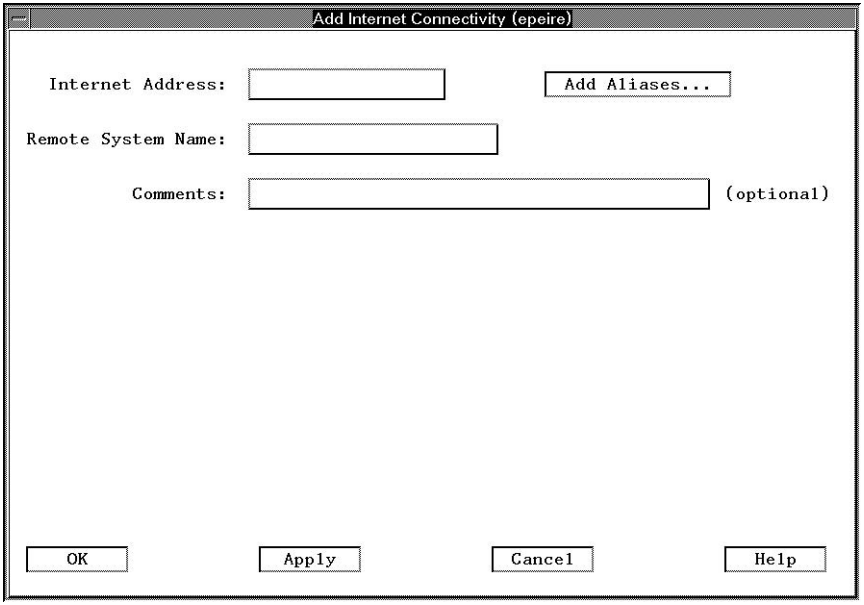
Step 3. At the "Networking and Communications" window, highlight `Hosts` and click on `OPEN`. Highlight `Local Hosts` and click on `OPEN`.

SMH displays all remote system names and IP addresses that have already been configured.

Step 4. Select `Add` from the `Actions` menu. The following dialog appears:

NOTE The appearance of the "Add Internet Connectivity" dialog may be slightly different from the example below depending on the version of the X.25 software you are running.

Figure 3-9 Add Internet Connectivity Dialog



Step 5. Enter or modify field values as required (refer to the field descriptions below). Use the `Help` button for information and instructions for each field.

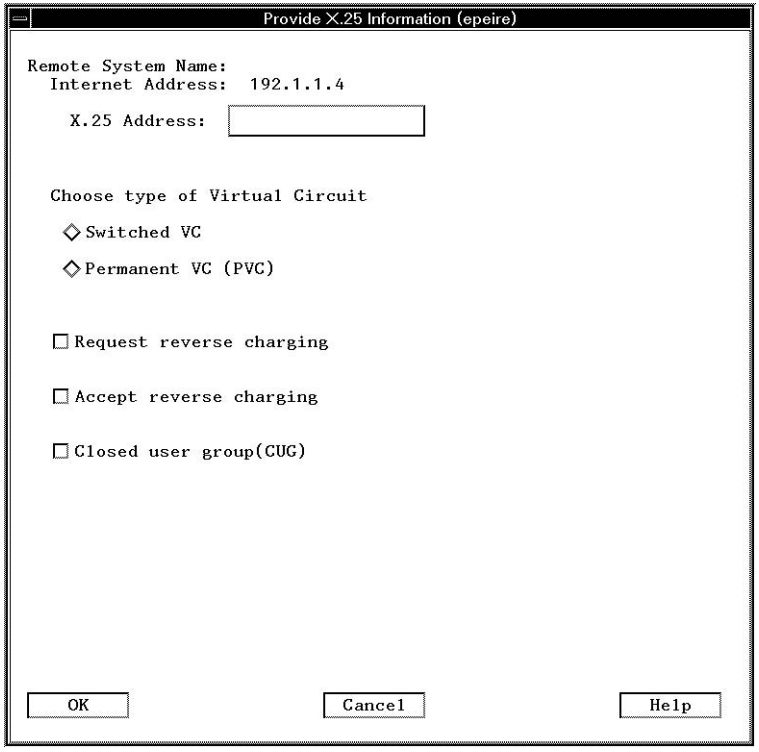
A `Provide X.25 Information` button may appear on this dialog depending on the Internet address you configure. If it does, follow the instructions in Step 6. SMH may also decide that a gateway is required, again depending on the Internet address you configure (see the on-line Help for more information on gateways and the gateway dialog).

Table 3-12 Add Internet Connectivity Field Descriptions

Internet Address	Identifier by which the remote system is known on the network. It is composed of four integers (0 to 255) separated by periods (for example, 192.2.3.6). Use the `Add Aliases` button to assign one or more aliases (in alphanumeric format) to the IP Address for easier referencing. When you exit this field, SMH determines whether a gateway is needed for the connection, or whether you need to provide X.25 information for the remote system. If the `Provide X.25 Information` button is displayed, follow the instructions in Step 6 below.
Remote System Name	Name assigned to the remote system to which you want to connect.
Comments	Use this field to add useful information about the remote system (for example, the name and telephone number of the user).

Step 6. If the `Provide X.25 Information` button is displayed, you must use it to configure X.25 information about the remote system. When you click on `Provide X.25 Information`, the following dialog appears:

Figure 3-10



Step 7. Enter or modify field values as required (refer to the field descriptions below).

If you add more VCs to an interface in the future, you must stop the interface with `x25stop` or with SMH's `Disable Card` function and restart it as an IP interface with SMH's `Enable Card` function or with the following command:

`x25init -c [`*`configuration_file`*`]-a [`*`ipmapfile`*`]`

Table 3-13 Provide X.25 Information Field Descriptions

X25 Address	X.25 address (sometimes referred to as the X.121 address) of the remote system. The X.25 address assigned by the network provider uniquely identifies the node in an X.25 network. The address consists of a maximum of 15 digits. You must complete this field.
Switched VC	Enable this field if a switched virtual circuit (SVC) will be used for this connection. When Switched VC is enabled (default), three additional fields can be configured: Request reverse charging, Accept reverse charging, and Closed user group (refer to the field descriptions below).
Permanent VC	Enable this field if a permanent virtual circuit (PVC) will be used for this connection. When Permanent VC is enabled, two additional fields can be configured: PVC Number and Local Programmatic Access Name.
Request reverse charging	Enable this field if the local system will make collect calls to the remote system. If this field is disabled, no collect calls can be made.
Accept reverse charging	Enable this field if you want the local system to accept collect calls from the remote system. If this field is disabled, no collect calls will be accepted.
Closed user group	Enable this field if you subscribed to a closed user group (CUG) and you want to use the CUG with this connection. Do not select this item if you did not subscribe to a CUG or if you do not want this connection to belong to a CUG. If you enable this field, an additional field (CUG Number) appears. Enter the number of the CUG to be associated with this connection.

Step 8. Click on `Apply` or `OK`. The `Apply` button leaves you in the current dialog so you can configure other remote systems; the `OK` button returns you to the `Internet Addresses (& Routes)` object list. SMH updates the object list to include the remote system you configured regardless of which button you used.

NOTE You can modify or remove remote systems and modify default gateways by highlighting the `Remote System Name` on the object list and selecting `Modify, Remove`, or `Modify Default Gateway` from the `Actions` menu.

Step 9. Select `Exit` from the `File` menu.

Remote system access for your X.25 link is now configured.

Configuring X.25 over LLC2

This section describes how to configure X.25 over LLC2. Configuring X.25 over LLC2 allows X.25 layer 3 to connect to DLPI instead of LAP-B and provides the X.25 functionality on top of LAN cards such as Ethernet, FDDI, and TokenRing.

NOTE In this document, an XOL interface on the peer or local system is referred to as “box”.

Configuration Files

To configure the LLC2 feature, the following configuration files must be added or updated manually:

- Generic X.25 configuration file
- LLC2 specific configuration file

Generic X.25 Configuration file

The generic X.25 configuration file is used with a few additional XOL specific parameters to configure X.25 over LLC2 (XOL). To support the XOL HA feature, some of the existing parameter names are modified and a few new parameters are introduced.

NOTE The parameters introduced in an earlier release of X.25 are supported. However, HP recommends that you migrate to the new parameters. Migration is required for XOL HA support.

Table 3-14 lists the XOL specific parameters that must be defined while configuring an XOL interface.

Table 3-14 Additional XOL specific configuration parameters

Field	Format	Value
device	Device name string.	Name of the LAN card where the XOL interface is configured.
XOL_logical_port_id (replaces lan_box_id)	Integer 0-n	The XOL logical port identifier for the XOL interface on the given LAN card.
XOL_destination_macaddr (replaces the MAC address in the lan_box_addr)	One hexadecimal value. This value must be prefixed with 0x.	The destination MAC address to which the XOL interface is connected.
XOL_DSAP	One hexadecimal value. This value must be prefixed with 0x.	The destination SAP value of the peer XOL interface. Set the value of DSAP to any even value in the range 0x70 - 0x7e, both inclusive.
XOL_SSAP	One hexadecimal value. This value must be prefixed with 0x.	The source SAP value of the local XOL interface. Set the value of SSAP to any even value in the range 0x70 - 0x7e, both inclusive. If the value for SSAP is not provided, then SSAP takes the value of DSAP.

Table 3-14 Additional XOL specific configuration parameters (Continued)

Field	Format	Value
standby_device**	Device name string	Name of the standby LAN device used for local failover. For more details about this parameter, see "Configuring the High Availability Feature for X.25 over LLC2" on page 81. Required only for local failover support.
XOL_floating_macaddr**	One hexadecimal value; must be prefixed with 0x	The unique MAC address for the LAN card in the LAN subnet. For more details about this parameter, see "Configuring the High Availability Feature for X.25 over LLC2" on page 81. Required only for local failover support.

** Required for HA feature

NOTE The configuration parameters may be specified in the old format. However, HP recommends that you use the new format while specifying the configuration parameter. XOL is not supported over APA (Auto Port Aggregation).

Following are two sample files. Example 3-1 provides a sample file that has the old parameters in the old format. Example 3-2 provides a sample file that has the new parameters in the new format.

Example 3-1 Sample File with the Old Parameters in the Old Format

```
device          lan00x7c
lan_box_id      1
lan_box_addr    0x080009c4728a 0x7e
```

Example 3-2 Sample File with the New Parameters in the New Format

```
device          lan0
XOL_logical_port_id     1
XOL_destination_macaddr 0x080009c4728a
XOL_DSAP 0x72
XOL_SSAP 0x7C
standby_device lan1
XOL_floating_macaddr 0x0060B0A4EBE5
```

A sample generic configuration file is available for reference. It is available in the `/etc/x25` directory and the filename is `x25init_llc2_smpl`.

Specific Configuration File

This file is an X.25 over LLC2 specific configuration file and contains information about every LAN. This file is unique to the host.

Following is the syntax of this file:

```
lan<#>     <number of boxes>
```

Following is a sample configuration file:

```
lan0        3
lan1        1
```

Following is an example of an XOL configuration file that is installed in `/etc/x25` directory. Following are the assumptions: The X.121 address of the local XOL interface is 7111 and that of the remote XOL interface is 7222.

Example 3-3 LLC2 Configuration

```
#
# Likely runstring: x25init -c x25init_llc2_smpl
X.121 7111              # X.121 address
```

```
X.121_packet  ''  “     # use a null (i.e. length 0) packet address
(TransPacaddressing)
name interface0       # interface name for Level 3 access
device lan0              # device to initialize


XOL_logical_port_id 0            # logical port id for lan interface
XOL_DSAP 0x7E                   # destination SAP id
XOL_SSAP 0x7E                   # source SAP id
XOL_destination_macaddr 0x080009c4728a  # Remote mac address
# Following needed only for Local High Availability feature
#standby_device lan1 # standby lan card for local failover.
#XOL_floating_macaddr 0x0060B0A4EBE5  # Unique Mac address to be assigned to the
lan card before bringing the XOL interface up.
# Level 2 Parameters
#
t1      3000
t3      60000
framesize       149
n2      20
l2window        7
# Level 3 Parameters
#       virtual circuit parameters
#       logical channel id,  start num [1-4095],  type,  how many
lci 1 pvc 5                       # 5 permanent VCs
lci 255 insvc 5                 # 5 one-way incoming SVCs
lci 2048 svc 6                  # 6 two-way switched VCs
networktype TRANSPAC    # CCITT 1984, DTE (see /etc/x25/x25_networks)
fast_select  enabled            # allow incoming calls with call user data
reverse_charge enabled     # allow incoming calls requesting reverse changes
def_inpacketsize 128     # default packetsize
def_outpacketsize 128   # default packetsize
```

```
def_inwindow 7          # default window size
def_outwindow 7         # default window size
def_inthruputclass 19200        # default thruput class
def_outthruputclass 19200       # default thruput class
flowcontrol  on               # flow control negotiation allowed
neg_inpacketsize 128    # offered packet size if using flow control negotiation
neg_outpacketsize 128  # offered packetsize if using flow control negotiation
neg_inwindow 7            # offered window size if using flow control negotiation
neg_outwindow 7 # offered window size if using flow control negotiation
thruputclass on
neg_inthruputclass 19200        # offered thruput class if using thruput class
negotiation
neg_outthruputclass 19200       # offered thruput class if using thruput class
negotiation
pvc_inpacketsize 128            # packetsize for PVCs
pvc_outpacketsize 128           # packetsize for PVCs
pvc_inwindow 7                  # window size for PVCs
pvc_outwindow 7                 # window size for PVCs
# IP Related Parameters
IP  15.4.64.120  255.255.248.0  # IP address and subnet mask
mtu 1024                        # max transmission unit 1024 octets
hold 300                        # 5 minute hold timer
idle 600                        # 10 minute idle timer
```

NOTE For an XOL configuration, L3 packet sizes greater than 1024 are not supported.

If the XOL interface on the host is connected to a router supporting X.25 over LLC2, the router must be configured with the correct routing entries. For example, if an XOL interface on the host is connected to a CISCO router and the XOL configuration on the host is identical to the configuration in the previous example, the following configuration is required:

- To enable the cmns, complete the following steps:

Step 1. Run the following command at the CISCO console:

```
Configure Terminal
```

Step 2. Run the following command at the CISCO console:

```
Interface fastethernet 1/0
```

Step 3. Run the following command at the CISCO console:

```
cmns enable
```

Step 4. Run the following command at the CISCO console:

```
<Ctrl+Z>
```

- To add routing entries to X.25 routing table, complete the following steps:

Step 1. Run the following command at the CISCO console:

```
Configure Terminal
```

Step 2. Run the following command at the CISCO console:

```
x25 route ^ 7111 interface FastEthernet1/10 mac
0011.0a80.31d6
```

Step 3. Run the following command at the CISCO console:

```
x25 route ^ 7222 interface FastEthernet1/10 mac
0800.09c4.728a
```

Step 4. Run the following command at the CISCO console:

```
<Ctrl+Z>
```

Starting and Stopping Configuration

Starting

To initialize the x25overllc2 devices, execute the x25init command with the -L option at the command prompt.

The syntax for execution is:

```
x25init -c <config_file> -L <llc2_config_file>
```

config_file is the generic X.25 product configuration file and llc2_config_file is the LLC2 specific configuration file.

A sample invocation is as shown below:

```
x25init -c x25init_def -L llc2_conf_def
```

This command is executed once per box. It can be executed with different <x25init_llc2_sampl> file but with the same <llc2_conf_def> file.

Stopping

To stop the communication with one box, the x25stop command is executed with the -L option.The syntax for execution is:

```
x25stop -d <lan#> -L <XOL_logical_port_id>
```

<XOL_logical_port_id> is the XOL logical port identifier for the XOL interface on the given LAN card.

NOTE SMH support is not available for configuring X.25 over LAN(LLC2). Also, the x25init command needs to be executed manually if the system is rebooted.

For a detailed description of the configuration, see *x25overllc2* (7). For more information on how to configure High Availability, see “Configuring the High Availability Feature” on page 76.

For more information on LLC2, see *x25init* (1M), *x25stop* (1M), and *x25stat* (1M) manpages.

NOTE To configure a Highly Available XOL interface, See “X.25 High Availability Configuration” on page 78.

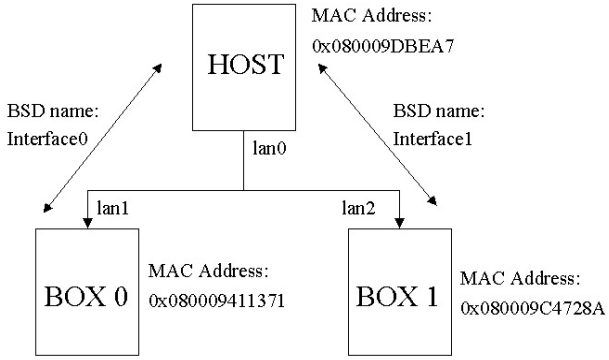
Sample setup and Configuration

The X.25 over llc2 supports one or more XOL interfaces. Following are the two example configurations:

One Lan and Two Boxes configuration

A graphical representation of the above configuration with some arbitrary values for the MAC addresses and lan-ids is shown below:

Figure 3-11 One LAN, Two Boxes Configuration



To connect HOST to BOX 0

To implement the above configuration, the following configuration files need to be created on the HOST system.

- Generic X.25 configuration file, x25_host_box_0
- LLC2 configuration file, llc2_lan_0

The generic X25 configuration file has to be created in the /etc/x25 directory using the example configuration file, /etc/x25/x25init_def.

The file x25_host_box_0 should contain all the mandatory level 2 and level 3 parameters. Along with them it should contain the following entries:

```
device lan 0

XOL_logical_port_id 0

XOL_destination_macaddr 0x080009411371

XOL_DSAP 0x70

XOL_SSAP 0x70
```

The LLC2 specific configuration file, llc2_lan_0 is created in the /etc/x25 directory with the following entries:

```
lan0            2
```

To connect HOST to BOX 1

To configure HOST to communicate with BOX1, the generic configuration file named `/etc/x25/x25_host_box_1` has be created as follows:

Apart from the mandatory level2 and level3 parameters the following parameters have to be added:

```
device lan 0

XOL_logical_port_id 1

XOL_destination_macaddr 0x080009C4728A

XOL_DSAP 0x7E

XOL_SSAP 0x7E
```

The LLC2 specific configuration is `/etc/x25/llc2_lan_0`.

To start and stop above configuration:

Starting:

Execute the following commands at the command prompt:

```
x25init -c x25_host_box_0 -L llc2_lan_0
```

Stopping:

To stop the configuration execute the following commands:

```
x25stop -d lan0 -L 0 # stop XOL interface with logical port id 0 on lan0
x25stop -d lan0 -L 1 # stop XOL interface with logical port id 1 on lan0
```

To Connect Box 0 to HOST

To implement the above configuration, the following configuration files need to be created on the BOX 0.

- Generic X.25 configuration file, x25_box_0_host
- LLC2 configuration file, llc2_lan_1

The generic X25 configuration file has to be created in the `/etc/x25` directory using the example configuration file, `/etc/x25/x25init_llc2_smpl`.

```
device lan1

XOL_logical_port_id 0

XOL_destination_macaddr 0x080009DBEA7

XOL_DSAP 0x70

XOL_SSAP 0x70
```

The file x25_box_0_host must contain all the mandatory level 2 and level 3 parameters. Along with them it should contain the following entries:

The LLC2 specific configuration file, llc2_lan_1 is created in the /etc/x25 directory with the following entries:

```
  lan1          1
```

To start and stop above configuration:

Starting:

Execute the following commands at the command prompt:

```
x25init -c x25_box_0_host -L llc2_lan_1
```

Stopping:

To stop the configuration execute the following commands:

```
x25stop -d lan1 -L 0 # stop XOL interface with logical port id 0 on lan1
```

To Connect Box 1 to HOST

To implement the above configuration, the following configuration files need to be created on the BOX 1.

- Generic X.25 configuration file, x25_box_1_host
- LLC2 configuration file, llc2_lan_2

The generic X25 configuration file has to be created in the `/etc/x25` directory using the example configuration file, `/etc/x25/x25init_llc2_smpl`.

The file `x25_box_1_host` must contain all the mandatory level 2 and level 3 parameters. Along with them it must contain the following entries:

```
device lan2
XOL_logical_port_id 0
XOL_destination_macaddr 0x080009DBEA7
XOL_DSAP 0x7E
XOL_SSAP 0x7E
```

The LLC2 specific configuration file, llc2_lan_2 is created in the `/etc/x25` directory with the following entries:

```
lan2            1
```

To start and stop above configuration:

Starting:

Execute the following commands at the command prompt:

```
x25init -c x25_box_1_host -L llc2_lan_2
```

Stopping:

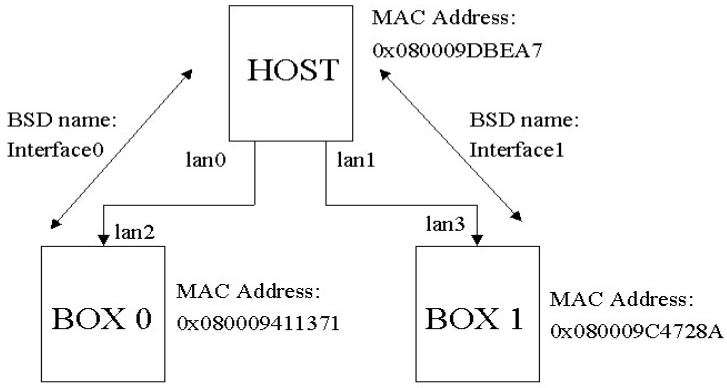
To stop the configuration execute the following commands:

```
x25stop -d lan2 -L 0 # stop XOL interface with logical port id 0 on lan2
```

Configuring Two Lans, with One Box Per Lan

A graphical representation of the above configuration with some arbitrary values for the MAC addresses and lan-ids is as shown:

Figure 3-12 Two Lans, One Box per Lan configuration



To Configure Host to LAN0 to Box0

To configure the Host to LAN 0, BOX 0 communication the following configuration files need to be created.

- Generic X.25 configuration file, x25_lan_0_box_0
- LLC2 configuration file, llc2_lan_0_lan_1

The x25_lan-0_box_0 file will contain all the level 3 and level2 parameters. It also contains the following entries:

```
device lan0
XOL_logical_port_id 1
XOL_destination_macaddr 0x080009411371
XOL_DSAP 0x70
XOL_SSAP 0x70
```

The LLC2 configuration file, llc2_lan_0_lan_1 will have the following entires:

```
lan0        1
lan1        1
```

To configure Host to LAN 1 to BOX 1

To configure the Host to LAN 1 to BOX 1communication, the following configuration file need to be created.

- X.25 generic configuration file, x25_lan_1_box_1
- LLC2 configuration file, llc2_2_lan_0_lan_1

The x25_lan_1_box_1 file is created in the `/etc/x25` directory. It will contain all the level 2 and level 3 parameters. It will also contain the following entries:

```
device lan1

XOL_logical_port_id 1

XOL_destination_macaddr 0x080009C4728A

XOL_DSAP 0x7E

XOL_SSAP 0x7E
```

There is only one LLC2 specific configuration file per Host. The file is llc2_lan_0_lan_1.

To Start and Stop above configuration:

Starting:

After reconfiguring the kernel of the two boxes execute the following commands at the command prompt:

```
x25init -c x25_lan_0_box_0 -L llc2_lan_0_lan_1
x25init -c x25_lan_1_box_1 -L llc2_lan_0_lan_1
```

Stopping:

To stop the configuration, execute the following commands at the command prompt:

```
xstop -d lan0 -L 1 # stop XOL interface with logical port id 1 on lan0
xstop -d lan1 -L 1 # stop XOL interface with logical port id 1 on lan1
```

To Connect Box 0 to HOST

To implement the above configuration, the following configuration files need to be created on the BOX 0.

- Generic X.25 configuration file, x25_box_0_host.
- LLC2 configuration file, llc2_lan_2

The generic X25 configuration file has to be created in the `/etc/x25` directory using the example configuration file, `/etc/x25/x25init_llc2_smpl`.

The file x25_box_0_host must contain all the mandatory level 2 and level 3 parameters. Along with them it must contain the following entries:

```
device lan2

XOL_logical_port_id 0

XOL_destination_macaddr 0x080009DBEA7

XOL_DSAP 0x70

XOL_SSAP 0x70
```

The LLC2 specific configuration file, llc2_lan_2 is created in the /etc/x25 directory with the following entries:

```
lan2            1
```

To start and stop above configuration:

Starting:

Execute the following commands at the command prompt:

```
x25init -c x25_box_0_host -L llc2_lan_2
```

Stopping:

To stop the configuration execute the following commands:

```
x25stop -d lan2 -L 0 # stop XOL interface with logical port id 0 on lan2
```

To Connect Box 1 to HOST

To implement the above configuration, the following configuration files need to be created on the BOX 1.

- Generic X.25 configuration file, x25_box_1_host.
- LLC2 configuration file, llc2_lan_3

The generic X25 configuration file has to be created in the `/etc/x25` directory using the example configuration file, `/etc/x25/x25init_llc2_smpl`.

The file x25_box_0_host should contain all the mandatory level 2 and level 3 parameters. Along with them it should contain the following entries:

```
device lan3

XOL_logical_port_id 0

XOL_destination_macaddr 0x080009DBEA7

XOL_DSAP 0x7E

XOL_SSAP 0x7E
```

The LLC2 specific configuration file, llc2_lan_2 is created in the /etc/x25 directory with the following entries:

```
    lan3          1
```

To start and stop above configuration:

Starting:

Execute the following commands at the command prompt:

```
x25init -c x25_box_0_host -L llc2_lan_3
```

Stopping:

To stop the configuration execute the following commands:

```
x25stop -d lan3 -L 0 # stop XOL interface having logical port id 0 on lan3
```

Troubleshooting LLC2 Configuration

Symptom: Unable to connect to the remote node.

Causes:

- Hardware address of the remote node is wrongly configured in the configuration file.
- The SAP value is incorrectly configured.
- The LCI value is incorrectly configured.

Action:

- Check the remote node's hardware address using lanscan. Use the same hardware address in the generic X.25 configuration file. If you modify the hardware address to correct the error then, re-initialize the card using x25init.
- If the nodes are connected back to back, check if the SAP's are identical for both the nodes in the generic X.25 configuration file.
- If the nodes are connected back to back and one of the machines is configured for inbound connections only and the other machine is configured for outbound connections only then, the starting LCI for both the nodes must be identical in the generic X.25 configuration file.

Configuring PAD Services

This section contains step-by-step instructions for configuring PAD services. Skip this section if you don't need to configure PAD services.

The procedures in this section are optional. Your X.25 link software does not require you to configure PAD services.

You can use SMH to add or modify the following:

- PAD terminal emulation (local to remote)
- PAD support server (remote to local)
- PAD printer server
- PAD UUCP server
- X.3 sets of parameters

See Chapter 6, PAD Services, for detailed reference information on PAD services.

This section assumes that SMH is running and that you are starting from the main window. If this is not the case, type `smh` at the HP-UX prompt before continuing.

Remember to use the `Help` button (in the SMH window) to display information and instructions for the content of each field.

Add/Modify PAD Terminal Emulation (Local to Remote)

Follow the steps below to add or modify PAD terminal emulation for a remote system:

Step 1. At the SMH main window, highlight "Networking and Communications" and select `Open Item` from the `Actions` menu.

Step 2. At the "Networking and Communications" window, highlight `X.25 PAD Services` and select `Open Item` from the `Actions` menu.

Step 3. Highlight `Terminal Emulator` and select `Open Item` from the `Actions` menu.

SMH displays an object list that shows all remote systems currently configured for PAD terminal emulation.

Step 4. Select Add from the Actions menu to configure PAD terminal emulation for an unlisted remote system, or highlight a remote system from the list and select Modify.

Step 5. Enter or modify field values as required.

Step 6. Click on Apply to apply the new settings and to add additional system connections. Click on OK to save changes and exit this dialog.

Step 7. Select Exit from the List menu to return to the "X.25 PAD Services" dialog.

NOTE If you enabled Reverse Charge Requested, you must also configure the PAD support server. See "Add/Modify PAD Support Server (Remote to Local)" for step-by-step instructions for configuring the PAD support server.

Step 8. Select the next PAD service you need to configure from the list on the "X.25 PAD Services" dialog.

Add/Modify PAD Support Server (Remote to Local)

Follow the steps below to add or modify a PAD support server for a local terminal or system:

Step 1. At the SMH main window, highlight "Networking and Communications" and select Open Item from the Actions menu.

Step 2. At the "Networking and Communications" window, highlight X.25 PAD Services and select Open Item from the Actions menu.

Step 3. Highlight Support Server and select Open Item from the Actions menu.

SMH displays an object list that shows all terminals/systems currently configured for access via X.25.

Step 4. Select Add from the Actions menu to configure a terminal/system that is not listed for access via X.25, or highlight a terminal/system from the list and select Modify.

Step 5. Enter or modify field values as required.

Step 6. Click on `Apply` to apply the new settings and to add additional PAD support servers. Click on `OK` to save changes and exit this dialog.

Step 7. Select `Exit` from the List menu to return to the "X.25 PAD Services" dialog.

Step 8. Select the next PAD service you need to configure from the list on the "X.25 PAD Services" dialog.

Add/Modify PAD Printers Server

Follow the steps below to add or modify remote PAD printers:

Step 1. At the SMH main window, highlight "Networking and Communications" and select `Open Item` from the `Actions` menu.

Step 2. At the "Networking and Communications" window, highlight `X.25 PAD Services` and select `Open Item` from the `Actions` menu.

Step 3. Highlight `Printers Server` and select `Open Item` from the `Actions` menu.

SMH lists all remote printers that your system can currently access via X.25.

Step 4. Select `Add` from the `Actions` menu to configure access to an unlisted remote printer, or highlight a remote printer from the list and select `Modify`.

Step 5. Enter or modify field values as required.

Step 6. Click on `Apply` to apply the new settings and to add additional PAD printers. Click on `OK` to save changes and exit this dialog.

Step 7. Select `Exit` from the `List` menu to return to the "X.25 PAD Services" dialog.

Step 8. Select the next PAD service you need to configure from the list on the "X.25 PAD Services" dialog.

Add/Modify UUCP Server

Follow the steps below to add or modify a UUCP server:

Step 1. At the SMH main window, highlight "Networking and Communications" and select `Open Item` from the `Actions` menu.

Step 2. At the "Networking and Communications" window, highlight `X.25 PAD Services` and select `Open Item` from the `Actions` menu.

Step 3. Highlight `UUCP Server` and select `Open Item` from the `Actions` menu.

SMH lists all remote systems currently configured for X.25 PAD UUCP connectivity.

Step 4. Select `Add` from the `Actions` menu to configure PAD UUCP connectivity for an unlisted remote system, or highlight a remote system from the list and select `Modify`.

Step 5. Enter or modify field values as required.

Step 6. Click on `Apply` to apply the new settings and to add additional PAD UUCP connections. Click on `OK` to save changes and exit this dialog.

Step 7. Select `Exit` from the `List` menu to return to the "X.25 PAD Services" dialog.

Step 8. If necessary, select the next PAD service you need to configure from the list.

In order for PAD UUCP connectivity to function correctly, UUCP must also be configured. If you have not already done this, you can configure UUCP by returning to the "Networking and Communications" window, highlighting `UUCP` and selecting `Open Item` from the `Actions` menu. From this dialog you can configure UUCP Devices and/or UUCP Remote Systems (see the on-line Help or UUCP documentation for more information).

Add/Modify X.3 Parameters

Follow the steps below to add or modify X.3 parameters. Refer to Chapter 6, PAD Services, for detailed descriptions of the X.3 parameters.

NOTE The X.3 values that you configure here are *initially* downloaded to the PAD. As the terminal characteristics of your session change, the X.25 software automatically downloads new values for certain parameters. Do not be alarmed if the values of some parameters are different from the values you initially specified.

Step 1. At the SMH main window, highlight "Networking and Communications" and select `Open Item` from the `Actions` menu.

Step 2. At the "Networking and Communications" window, highlight `X.25 PAD Services` and select `Open Item` from the `Actions` menu.

Step 3. Highlight `X.3 Sets of Parameters` and select `Open Item` from the `Actions` menu.

SMH displays an object list showing all X.3 sets or profiles currently configured.

Step 4. Select `Add` from the `Actions` menu to configure an unlisted X.3 set, or highlight an X.3 set from the list and select `Modify`.

When you select `Add`, SMH displays the "Set X.3 Default Values" dialog. This dialog lets you set defaults for PAD terminal emulation, PAD support server, PAD printers server, or PAD UUCP server.

Step 5. Enter or modify field values as required. Click on `Help` for information and instructions for each field. A full description of X.3 parameters are provided in Chapter 6, "PAD Services."

Step 6. Click on `Apply` to apply the new settings and to add additional X.3 sets. Click on `OK` to save changes and exit this dialog.

Step 7. Select `Exit` from the `List` menu.

Configuring the High Availability Feature

The X.25 link software provides integration of the X.25/Streams solution to the High Availability HP MC/ServiceGuard feature on HP 9000 Series 800 Systems, allowing local and remote failure recovery.

This feature requires that the HP MC/ServiceGuard product (B3936AA) has been installed and that the X.25 software uses switches that support Hunt Group Facility allowing multiple DTEs to share a common X.121 address.

You should be familiar with the HP MC/ServiceGuard product prior to installation. For more information on Serviceguard, see *Managing MC/ServiceGuard*.

The High Availability features include:

- **Local Failure:** Support of local failure for an X.25 link is provided by a PDN using existing X.25 functionality. If a local X.25 PSI card fails, the calling DTE will see the link go down.

  The Hunt Group facility results in all incoming calls being directed to the remaining “good” X.25 PSI card in the system.

- **Remote Failure:** HP MC/ServiceGuard, uses the `x25ifstate` command to monitor the X.25 status. If the monitor detects that the X.25 link it is monitoring has gone down, all it has to do is exit.

  HP MC/ServiceGuard detects that the monitor service belonging to the package has disappeared, stops the package, and starts the migration towards a second system.

  The X.121 address of the first package is taken over by the second system from the X.25 configuration file.

X.25 Cluster Definition

To configure a High Availability cluster, use SMH to define all cluster systems by giving their:

NODE_NAME

NETWORK_INTERFACE

HEARTBEAT_IP

X.25 Commands to be Configured in High Availability Packages

Start X.25 link

`x25init -c x25config -a ipmap -d x25dev` (see `x25init (1m)`)

Stop X.25 link

`x25stop -d x25dev` (see `x25stop (1m)`)

Monitor X.25 link

`x25ifstate -d x25dev -H` (see `x25ifstate (1m)`)

X.25 Package Definition

An X.25 package is needed for each HA X.25 node. This package allows HP MC/ServiceGuard to automatically:

- start the X.25 link (`x25init` command) when the X.25 package starts
- start a process (`x25ifstate` command) to monitor the state of the X.25 link
- stop the failed X.25 package before migrating to a backup node

To define each X.25 package, you need:

Package Name:

to identify the X.25 package. Use a name such as `x25pkg1`.

Service Name:

to monitor the X.25 link to be managed

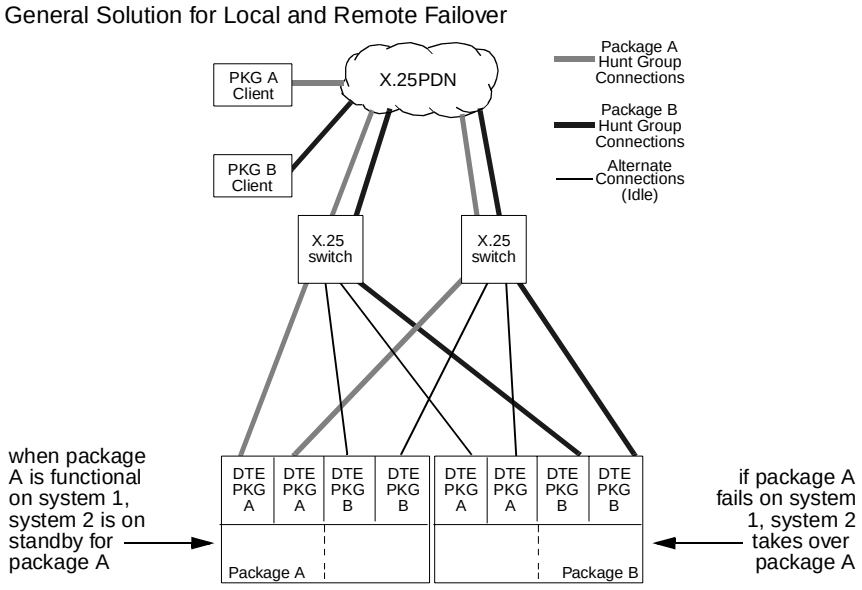
Service Command:

to monitor the X.25 link status. It could be either a script based on `x25ifstate` or the `x25ifstate` command directly, depending on your configuration.

Package Control Script Location:

location of the script to start/stop the X.25 link and execute the service(s).

X.25 High Availability Configuration



Example 3-4 One EXAMPLE of Remote Failure

The only file that contains the X.25 commands to run for High Availability is the `/etc./cmcluster/pkg*/control.sh` file (one per X.25 package).

```
# SERVICE NAMES AND COMMANDS
SERVICE_NAME [0] =pkg1.Service_Name_01 (same SERVICE_NAME as in pkg conf)
SERVICE_CMD [0] =”/usr/sbin/x25ifstate -d /dev/x25_1 -H -t 5”
SERVICE_RESTART [0] = “ “

#add below the service names which define the applications that rely on the X.25
card
```

```
# e.g.# SERVICE_NAME [1] = <other name>
# SERVICE_CMD [1] = <other monitor>
# START OF CUSTOMER DEFINED FUNCTIONS
function customer_defined_run_cmds
{
#ADD customer defined run commands
: # do nothing instruction, because a function must contain at least one command.
x25init -c /etc/x25/x25config_1 -a /etc/x25/ip_to_x121_map
sleep 5
test_return 51
}
function customer_defined_halt_cmds
{
#ADD customer defined halt commands
: # do nothing instruction, because a function must contain some command.
#stopping the specific X.25 card means the signal modem will be down
x25stop -d /dev/x25_1
test_return 52
}
```

Checking the Configuration

Check that your configuration is correct using `x25server`, `x25check`, and `ping`.

x25server Run the `x25server` command. This starts a background process that waits for packets from an `x25check` command and sends a reply.

x25check Execute the `x25check` command, using your X.121 address:

```
$ x25check 250207

X25CHECK (c) COPYRIGHT Hewlett-Packard Company 1988.
Test Starts on .. Wed Dec 8 09:15:01 1993

Initialization of the test...
CALL packet sent ...

The following figures have been measured on the network:
 Set up time : 267 ms
Remote Connection Succeeded
```

Execute the `x25check` command again, but this time with data packets:

```
$ x25check 250207 -s 32 -n 1

X25CHECK (c) COPYRIGHT Hewlett-Packard Company 1988.
Test Starts on .. Wed Dec 8 09:15:47 1993

Initialization of the test...
CALL packet sent ...
DATA packet sent ... DATA packet received

The following figures have been measured on the network:
Set up time : 286 ms
Transit time : 245 ms
Remote Connection Succeeded
```

If your system has multiple X.25 cards, you may need to specify the particular interface in order for `x25check` to work properly. For example, to run `x25check` on the second port of the first card (interface `x25_0p2`), you would use the following format:

```
x25check 250207 -i interface0p2
```

ping command Run the `ping` command to check your IP over X.25 connection. You can use either an IP address, or the alias for the local node (defined in the `hosts` file). The following example sends 6 packets of 100 bytes:

```
ping hpindla 100 -n 6

PING hpindla: 100 byte packets
100 bytes from 15.128.131.152: icmp_seq=1. time=321. ms
100 bytes from 15.128.131.152: icmp_seq=2. time=320. ms
100 bytes from 15.128.131.152: icmp_seq=3. time=319. ms
100 bytes from 15.128.131.152: icmp_seq=4. time=318. ms
100 bytes from 15.128.131.152: icmp_seq=5. time=320. ms
100 bytes from 15.128.131.152: icmp_seq=6. time=321. ms

--hpindla PING Statistics--
6 packets transmitted, 6 packets received, 0% packet loss
round-trip (ms) min/avg/max = 318/319/321
```

You can then check your IP over X.25 connection to a remote host using its IP address or alias.

If `ping` does not run successfully on the local node address, refer to Chapter 8, “Troubleshooting.”

Configuring the High Availability Feature for X.25 over LLC2

This section describes how to configure the local and remote failover features for X.25 over LLC2 (XOL) interface. It also describes the local, remote, and a combination of local and remote failover.

WARNING **The failover is not transparent and results in loss of data and existing connections. After a failover, the XOL connections must be re-established.**

Unique MAC Address Requirement

XOL High Availability (XOL HA) is based on sharing a unique MAC address for the primary and standby LAN cards, whichever is active. The use of a unique MAC address makes the card swap transparent from the XOL router side or peer system, which sees the same MAC address associated with the active interface.

NOTE To avoid MAC address collision, the system administrator must identify an unused, unique MAC address in their LAN subnet. The `lanadmin` command is used to change the MAC address of the card. After failover, the failed LAN card is reconfigured with its factory default MAC address. XOL is not supported over APA (Auto Port Aggregation).

Before you run the `lanadmin` command to set up the unique MAC address, you may test whether the MAC address chosen by you is selected by running the `linkloop` command. For more information, see `linkloop` (1m). The uniqueness of the MAC address is verified on the LAN cards currently active in the LAN segment. The `linkloop` command sends an OK message if the MAC address is already in use.

NOTE The routing table of the router connected to the host system must be updated with the unique MAC address identified by the user. If the XOL router supports the Hunt Group facility for XOL HA, the unique MAC address may not be required to achieve the HA functionality for XOL interface. For more information on Hunt Group facility, see "Configuring the High Availability Feature for X.25 over LLC2" on page 81.

WARNING **When several highly available XOL interfaces are configured on a given LAN card, failover of all the highly available XOL interfaces to the standby LAN card occurs immediately after one XOL interface fails. This is because the same unique MAC address is used by all XOL interfaces on the given LAN card.**

Local Failover

NOTE You need not install ServiceGuard for Local failover.

For local failover to happen, you must define a standby LAN card in the X.25 configuration file. The primary X.25 configuration file is used to configure the XOL interface on the standby LAN card after the local failover. As a result, the standby LAN card inherits all the configuration parameters from the primary interface, including the XOL logical port-id (also called box-id). The XOL interface starts first on the primary LAN card. If this XOL interface fails, an attempt is made to restart it on the standby LAN card. If the XOL interface fails on the standby LAN card, an attempt is made to failover back to the primary LAN card. This failover succeeds only if the primary LAN card is back UP and operational. Otherwise, the XOL interface goes down.

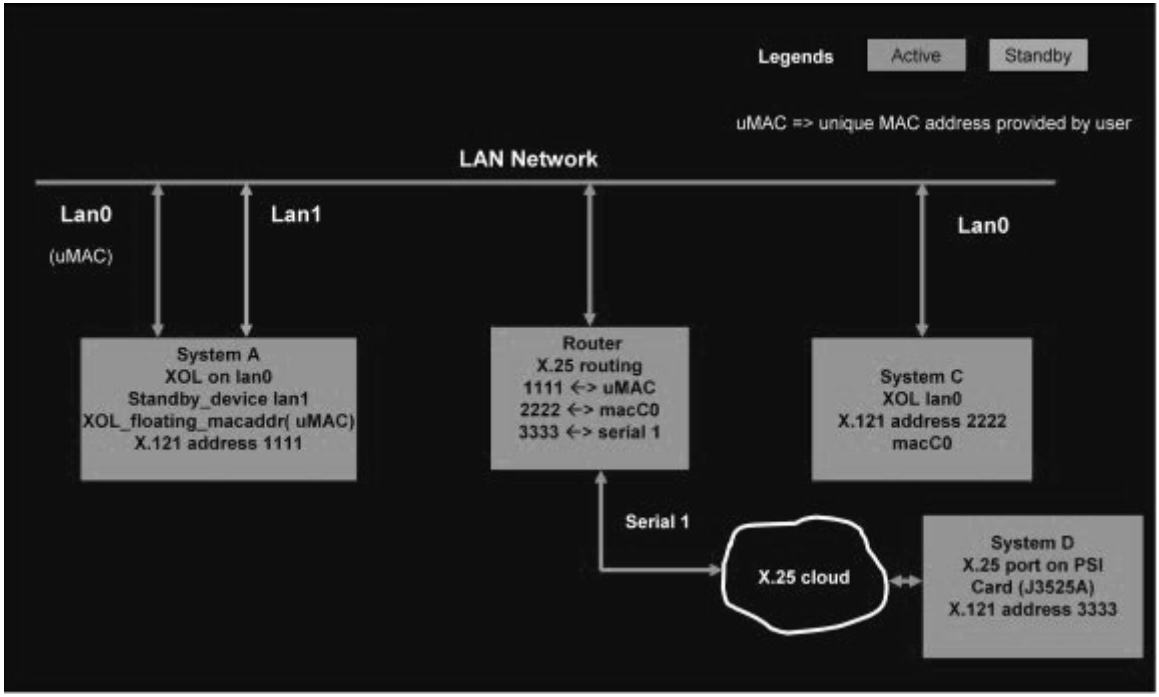
For more information on how to use this local failover functionality, see "Configuring XOL HA for Local Failover Without ServiceGuard" on page 83.

Configuring XOL HA for Local Failover Without ServiceGuard

In Figure 3-13, system A has two LAN cards, namely, lan0 (the primary LAN card) and lan1 (the standby LAN card). The standby LAN card provides local failover for the XOL interface with X.121 address 1111. The user supplied, unique MAC address that is specified in the XOL configuration file as a value for the field XOL_floating_macaddr, for that LAN segment is uMAC.

The XOL interface with X.121 address 1111 is brought up on lan0 on system A (uMAC is automatically assigned to lan0 on system A when the XOL interface is initialized).

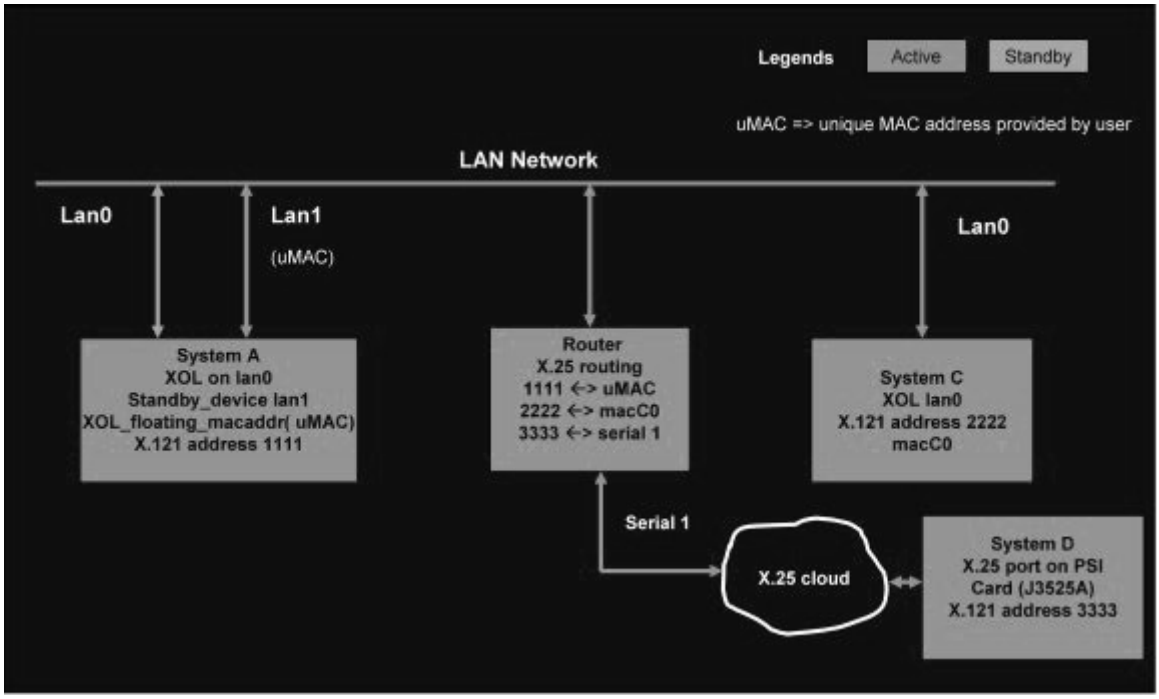
Figure 3-13 Before the Failover



When the lan0 (the primary LAN card) fails, the XOL interface stops on lan0, and a XOL interface is brought up automatically on lan1 (the standby LAN card on system A) using the same configuration file. An attempt is made to configure lan0 with the factory default MAC address, and uMAC is assigned to lan1 on system A during the local failover. The

X.25 connection must be re-established on the XOL interface. When the card lan0 comes up later, it is configured with the factory default MAC address.

Figure 3-14 After the Failover



To configure XOL local failover, complete the following steps:

NOTE The XOL HA feature is enabled by default in HP-UX 11i v3. Therefore, the -enableXOLHA option is no longer required.

Step 1. To configure the standby device and the unique MAC address in the configuration file of the XOL primary interface, modify the following syntax:

```
Standby_device <standby LAN card name> # used for local
failover

XOL_floating_macaddr <unique MAC address>
```

NOTE Local failover is automatically enabled by defining a standby_device in the configuration file. If you do not require the local failover feature for this interface, you must de-configure the standby_device by removing or commenting the line in the configuration file. The standby XOL interface inherits the configuration of the primary XOL interface during failover because the same configuration file is used to configure the XOL interface.

Example 3-5 Configuring Standby Device and Unique MAC Address

If the asumptions in the XOL configuration file are the following:

- The primary LAN card is lan0
- The standby LAN card is lan1
- The MAC address is 0x0060B0A4EbE5

Then, the XOL HA parameters take the following values:

```
standby_device  lan1

XOL_floating_macaddr  0x0060B0A4EBE5
```

NOTE In the event of a failover, the XOL_floating_macaddr is moved from the primary LAN card to the standby LAN card. HP recommends that you use the primary and standby LAN cards only for the XOL links configured with HA feature.

Step 2. To initialize the XOL HA interface with local failover support, run the following command:

```
x25init -c <XOL_config_file> -L <llc2_conf_def>
```

Remote Failover

NOTE Remote failover, with or without local failover, requires the installation of the ServiceGuard product on the system. For more information on HP MC/ServiceGuard product, see *Managing MC/ServiceGuard* document.

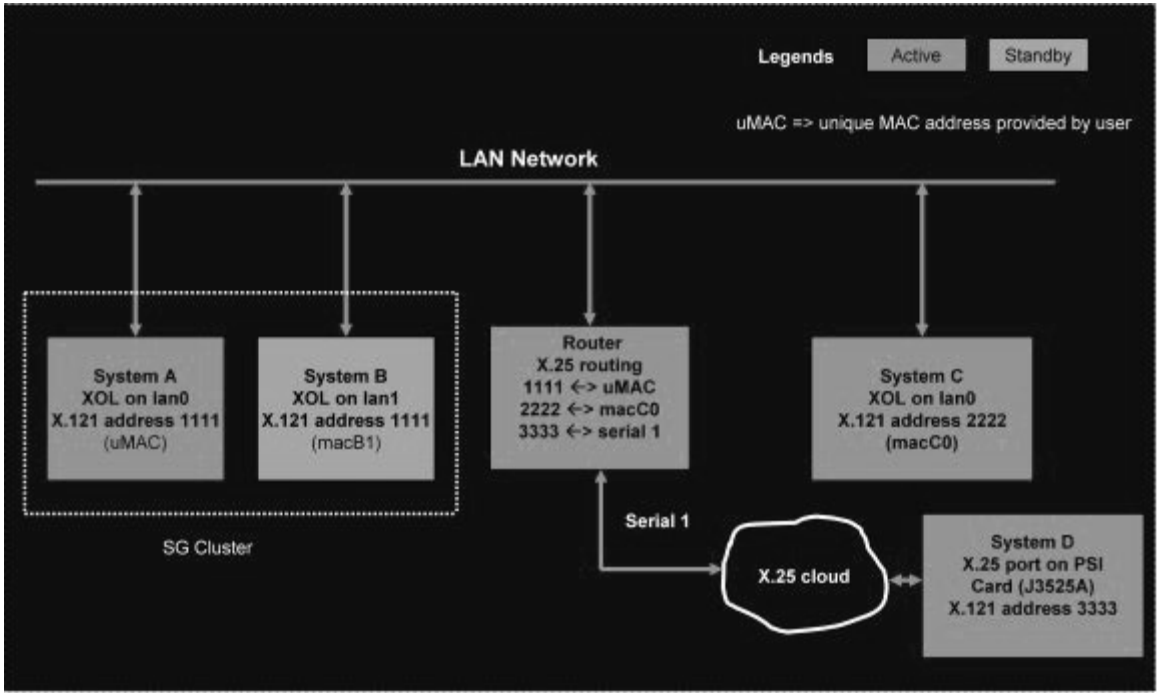
In the ServiceGuard cluster, the remote failover works between LAN cards on different nodes. Each LAN card has an XOL interface which is defined by the same configuration filename used in the SG package. The `x25ifstate` command monitors the XOL interface status. When the XOL interface fails, the `x25ifstate` command exits. HP MC/Service Guard detects that the monitor service belonging to the package has exited and stops the package. HP MC/Service Guard then starts the same package on the standby node of the ServiceGuard cluster, which starts the XOL interface on the LAN card on the standby node.

For more information on using the remote failover functionality with Service Guard, see “Configuring XOL HA for Remote Failover Using Service Guard (SG)” on page 86.

Configuring XOL HA for Remote Failover Using Service Guard (SG)

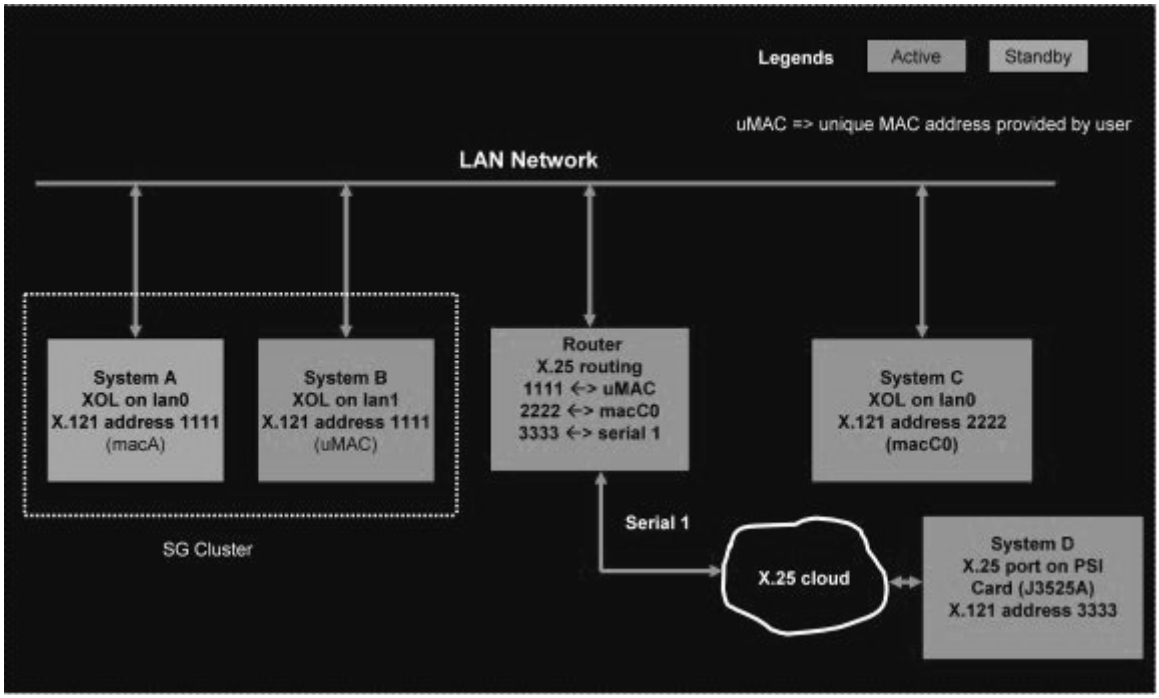
In Figure 3-15, primary system A and standby system B are part of the SG cluster and provide remote failover of an XOL interface with X.25 address 1111. The unique, user supplied MAC address is uMAC for that LAN segment. It is initially assigned to the card lan0 on primary system A.

Figure 3-15 Before the Failover



If the SG package for XOL detects an error in the XOL interface, the package is stopped on the primary system and the equivalent package starts on the standby system. When the SG package is stopped on system A, the XOL interfaces (and optionally applications) on the failed LAN card are stopped. An attempt is made to configure LAN card on system A with the factory default MAC address. When the SG package starts on system B, the LAN card on system B is configured with the unique MAC (uMAC) address, and the XOL interfaces (and optionally applications) starts on it. When the failed LAN card on system A comes up later, it is configured with the factory default MAC address.

Figure 3-16 After the Failover



XOL Cluster Definition

You must define the cluster by providing the system names that are part of SG cluster.

X.25 Commands Used in the SG Package Control Scripts:

- To set the MAC address, run the following command:

  ```
  lanadmin -A <MAC address>  <PPA>
  ```

- To start XOL interface, run the following command:

  ```
  x25init -c /etc/x25/x25config_llc2 -L
  /etc/x25/llc2_conf_def
  ```

- To monitor the XOL interface, run the following command:

  ```
  x25ifstate -d /dev/x25_0l1 -H
  ```

- To stop the XOL interface, run the following command:

```
x25stop -d /dev/x25_0l1
```

XOL Package Definition

An XOL SG package is required to configure a highly available XOL interface on all nodes in the SG cluster. This package allows HP MC/Service Guard to automatically:

- Assign a unique MAC address (using the lanadmin command) to the LAN card in use.
- Start the X.25 link (`x25init` command) when the X.25 package starts.
- Run the command `x25ifstate` to monitor the state of the XOL interface.
- Stop the XOL interface when the monitoring command detects a failure.

To define the X.25 package, you require the following attributes:

Package Name — Identifies the X.25 package, for example, xol_pkg1.

Service Name — Monitors the XOL interface to be managed, for example, xol_SG_service1

Service Command — Monitors the XOL interface status, for example, `x25ifstate`

To configure XOL remote failover, complete the following steps:

NOTE The XOL HA feature is enabled by default in HP-UX 11i v3. Therefore, the -enableXOLHA option is no longer required.

NOTE When the XOL interface goes down, the service command (`x25ifstate`) exits. The SG performs a package failover to the standby node configured in the SG cluster.

The SG package file that contains the X.25 commands is `/etc./cmcluster/pkg*/control.sh` file (one per XOL package). Following is a sample SG package file:

Example 3-6 XOL SG package for remote failover

```
# SERVICE NAMES AND COMMANDS
SERVICE_NAME [0] =pkg1.Service_Name_01 (same SERVICE_NAME as in pkg conf)
SERVICE_CMD [0] ="/usr/sbin/x25ifstate -d /dev/x25_0l1 -H "
SERVICE_RESTART [0] = " "

# Unique MAC address in the subnet (LAN segment), which is
# not in use. This will be the Active MAC address on the
# active node in the cluster.

Supplied_MAC = 0x0060B0A4EBE5

# START OF CUSTOMER DEFINED FUNCTIONS
function customer_defined_run_cmds
{
# START of customer defined run commands.
/usr/sbin/x25stop -d /dev/x25_0l1
sleep 2
/usr/sbin/lanadmin -A $Supplied_MAC 0
/usr/sbin/x25init -c /etc/x25/x25config_llc2_1 -L /etc/x25/llc2_conf_def
sleep 5
test_return 51
# END of customer defined run commands
```

```
}
function customer_defined_halt_cmds
{
# START of customer defined halt commands.
/usr/sbin/x25stop -d /dev/x25_0l1
Sleep 2
test_return 52
# END of customer defined halt commands
}
```

Remote Failover with Local Failover Support Using Service Guard

This is a combination of local failover and remote failover. For information on the local and remote failovers, see “Configuring XOL HA for Local Failover Without ServiceGuard” on page 83 and “Configuring XOL HA for Remote Failover Using Service Guard (SG)” on page 86. If the interface cannot be activated on either the primary or the standby LAN card at a given time on the local node, the SG package initiates the remote failover to the LAN card on the standby node in the SG cluster.

NOTE On any given LAN card, the combination of highly available and not highly available XOL interfaces is not supported. Either all or none of the interfaces must be configured with HA (for local or remote failover) on any given LAN card. This is because the same unique MAC address is used by all XOL interfaces on the given LAN card.

To support remote failover with local failover, complete the following steps:

NOTE Enabling or disabling of HA feature for XOL interface is system wide. Before enabling or disabling the HA feature on a system, all existing XOL interfaces must be stopped. The XOL HA feature is enabled by default on 11i v3.

Step 1. To configure the standby device and the unique MAC address in the configuration file of the XOL primary interface, modify the following syntax:

```
Standby_device <standby LAN card name> # used for local
failover

XOL_floating_macaddr <unique MAC address>
```

NOTE Local failover is automatically enabled if a “standby_device” is defined in the configuration file. If you do not want the local failover feature for this interface, you must de-configure the standby_device by removing or commenting the line in the configuration file. The XOL interface inherits the configuration of the primary XOL interface during failover because the same configuration file is used to configure the XOL interface.

Example 3-7 Configuring Standby Device and Unique MAC Address

If the assumptions in the XOL configuration file are the following:

- The primary LAN card is lan0
- The standby LAN card is lan1
- The MAC address is 0x0060B0A4EbE5

Then, the above parameters take the following values:

```
standby_device  lan1

XOL_floating_macaddr  0x0060B0A4EBE5
```

NOTE In the event of a failover, the XOL_floating_macaddr is moved from the Primary LAN card to the standby LAN card. HP recommends that you use the primary and standby LAN cards only for the XOL links configured with HA feature.

Step 2. Configure the SG package. For information on achieving remote failover functionality, see "XOL Package Definition" on page 89.
For remote failover with local failover support, define the XOL package with the following changes:

- In the Start script of the package, stop the monitor if it is running. For details, see Example 3-6 on page 90.
- In the XOL package definition, use `x25HAmonitor` command instead of `x25ifstate` as the service command, as follows:

```
x25HAmonitor -c /etc/x25/x25config_llc2 -L
/etc/x25/llc2_conf_def -m
```

NOTE The `x25HAmonitor` command helps in failover of the XOL interface between the primary and standby LAN cards (on the local system), if the LAN card (either primary or standby) on which the XOL interface was active fails. The `x25HAmonitor` exits if it is does not start the XOL interface on the local system on either the primary or standby LAN card. The SG does a package failover to the standby node configured in the SG cluster.

Step 3. To start the SG package, run the following command

```
cmrunpkg -v <package name>
```

Example 3-8 Remote Failover with Local Failover support

The following two entries must be added to the LLC2 configuration file:

```
standby_device lan2                              # lan2 is standby LAN.
XOL_floating_macaddr  0x0060B0A4EBE5  # Unique MAC address in the subnet
```

The SG package file that contains the X.25 commands is `/etc./cmcluster/pkg*/control.sh` file (one per XOL package).

```
# SERVICE NAMES AND COMMANDS
SERVICE_NAME [0] =pkg1.Service_Name_01 (same SERVICE_NAME as in pkg conf)
SERVICE_CMD [0] ="/usr/sbin/x25HAmonitor -c x25config_llc2 -L llc2_conf_def -m "
SERVICE_RESTART [0] = " "

# Unique MAC address in the subnet (LAN segment) which is not in use. This will
be the Active MAC address on the active node in the cluster.
Supplied_MAC = 0x0060B0A4EBE5

# START OF CUSTOMER DEFINED FUNCTIONS
function customer_defined_run_cmds
{
# START of customer defined run commands.
/usr/sbin/x25stop -d /dev/x25_1l0    # primary lan PPA is 1 and
XOL_logical_port_id is 0
sleep 2
/usr/sbin/lanadmin -A $Supplied_MAC 1
/usr/sbin/x25init -c /etc/x25/x25config_llc2_1 -L /etc/x25/llc2_conf_def
sleep 5
# Check and stop the x25HAmonitor if already running
# This is needed as Service Guard monitor starts the monitor later
  PID=`/usr/bin/ls /var/x25/tmp/x25HAmonitor_1l0.* | /usr/bin/awk -F '.' '{ print
$2 }'`
  PROC_NAME=`/usr/bin/ps -e | /usr/bin/grep $PID  /usr/bin/awk '{ print $4 }'`
        if [[ $PROC_NAME = "x25HAmonitor" ]]
        then
                /usr/bin/kill $PID
        fi
test_return 51
# END of customer defined run commands
}
```

```
function customer_defined_halt_cmds
{
# START of customer defined halt commands.
/usr/sbin/x25stop -d /dev/x25_1l0
Sleep 2
# Assuming that lan2 was configured as standby lan
/usr/sbin/x25stop -d /dev/x25_2l0
Sleep 2
test_return 52
# END of customer defined halt commands
}
```

Verifying the X.25 Link

After completing the installation and configuration instructions, follow these steps to verify your link:

Step 1. If you are using IP over X.25, use the `ping` command to check that the IP connection is operating properly. Refer to your man pages for details on the `ping` command. Examples are given below.

To check the connection to and from the network to the IP address (for example, 193.6.3.2), type:

```
ping 193.6.3.2
```

To check the full connection across the network to the remote system's IP address (for example, 193.6.1.1), type:

```
ping 193.6.1.1
```

Step 2. If `ping` was unsuccessful, use the `x25check` and `x25server` commands to verify that the X.25 connections are operating properly. Refer to your man pages for detailed information on the these commands.

NOTE If you have problems with the operation of your X.25 link, repeat the installation process described in Chapter 2, Installation, and the configuration instructions in this chapter, or refer to Chapter 8, Troubleshooting, for troubleshooting information.

4 OLA/R Overview and Concepts

Introduction

The letters O, L, A and R stand for On Line Addition [and] Replacement. This, of course, refers to the ability of a PCI I/O card to be replaced/added to an HP-UX computer system designed to support this feature without the need for completely shutting down, then re-booting the system or affecting other system components. The system hardware uses the per-slot power control combined with operating system support to enable this feature.

Initially, not all add-in cards will have this capability but over time users should see many cards adding this capability to their set of functions.

IMPORTANT Certain “Classes” of hardware are not intended for access by users. At this time this includes Z-class (SuperDome) systems. HP recommends that these systems only be opened by a qualified HP Engineer. Failure to observe this requirement can invalidate any support agreement or warantee to which the owner might otherwise be entitled.

IMPORTANT For those wishing to use OLAR, your system may need to update its system firmware. For additional details, please refer to the "Read Before Installing or Updating to HP-UX 11i v3" document.

Important Terms and Concepts

Table 4-1 Terms used in this section

Term	Meaning
OLA/R	All aspects of the OLA/R feature including On-line Addition (OLA) and On-line Replacement (OLR).
Power Domain	A grouping of 1 or more interface card slots that are powered on or off as a unit.
target card / target card slot	The interface card which will be added or replaced using OLA/R, and the card slot it resides in.
affected card / affected card slot	Interface cards and the card slots they reside in and are in the same power domain as the target slot.

IMPORTANT

In many cases, other interface cards and slots within the system are dependent upon the target card. For example:

- If the target card slot is in a power domain and you temporarily stop power to the target card slot, you will also stop power to any other card slots (affected card slots) in that power domain.
- If the target card is a multiple-function card (MFC), suspending or deleting drivers for the target card slot also suspends individual drivers for the multiple hardware paths on that card).

During a card replacement operation, SMH performs a *Critical Resource Analysis*, which checks the target card for critical resources that would be lost when the card is shut down.

Planning and Preparation

For the most part SMH prevents you from performing OLA/R procedures that would adversly affect other areas of the server. This section provides you with important information that can help minimize errors or problems when performing OLA/R procedures.

Card Compatibility

On-Line Addition

When on-line adding an interface card, the first issue that must be resolved is whether the new card is compatible with the system. Each OLA/R-capable PCI slot provides a set amount of power. The replacement card cannot draw more power than is available.

The card must also operate at the slot's bus frequency. A PCI card must run at any frequency lower than its maximum capability, but a card that could only operate at 33 MHz would not work on a bus running at 66 MHz. `rad` provides information about the bus frequency and power available at a slot, as well as other slot-related data.

On-Line Replacement

When on-line replacing an interface card, the replacement card must be identical to the card being replaced. This is referred to as *like-for-like* replacement and should be adhered to because using a similar but not identical card may cause unpredictable results. For example, a newer version of the target card which is identical in terms of hardware may contain an updated firmware version that could potentially conflict with the current driver.

The PCI specification allows a single physical card to contain more than one function. A single-function SCSI bus adapter can not be replaced by a dual-function adapter, even if the additional function on the card was identical to the original SCSI bus adapter.

When the replacement card is added to the system, the appropriate driver for that card must be configured in the kernel before beginning the operation. SMH ensures the correct driver is present. (In most cases, the replacement card will be the same type as a card already in the system, and this requirement will be automatically met.) If you have any

question about the driver's presence, or if you are not certain that the replacement card is identical to the existing card, you can use ioscan together with `rad` to investigate.

- During the replacement process, the original driver instance runs in a suspended state. I/O to the card is either queued or failed while the card is suspended. When the replacement card is brought on-line, the driver instance resumes normal operation. The driver instance must be capable of resuming and controlling the replacement card.
- If the necessary driver is not present and the driver is a dynamically loadable kernel module (DLKM), you can load it manually.
- If the driver is static and not configured in the kernel, then the card cannot be On-line Added. The card could be physically inserted on-line, but no driver would claim it.

Critical Resources

Replacing a card that is still operating can have extensive ramifications. Since power to the slot must be off when the old card is removed and the new card is inserted, the effects of shutting down the card's functions must be considered.

This is particularly important if there is no on-line failover or backup card to pick up those functions. For example:

- Which mass storage devices will be temporarily disconnected when the card is shut down?
- Will a critical networking connection be lost?

A critical resource is one that would cause a system crash or prevent the operation from successfully completing if the resource were temporarily suspended or disconnected. For example, if the SCSI adapter to be replaced connects to the unmirrored root disk or swap space, the system will crash when the card is shut down.

During an OLA/R procedure, it is essential to check the targeted card for critical resources, as well as the effects of existing disk mirrors and other situations where a card's functions can be taken over by another card that will not be affected.

Fortunately SMH performs a thorough critical resource analysis automatically, and presents options to you based on it's findings. If you determine that critical resources will be affected by the procedure, you

could replace the card when the server is off-line, or if you must take action immediately, you can use `rad` to attempt an on-line addition of a backup card and deletion of the target card.

Failover Actions / Single Points of Failure

In most cases, the system will automatically fail over to the alternate resource when a card is suspended. However, some subsystems might require manual intervention. For example, the Logical Volume Manager (LVM), will automatically redirect I/O for a temporarily disconnected disk resource to a mirror, logging errors as it handles this situation.

- Along those lines, if the resource will be suspended for an extended period of time, a large number of error log entries could result.
- In this type of situation, you may want to manually switch over to a mirror beforehand. When you have completed the OLA/R procedure, the mirror and disk can be resynchronized.

If you suspend a card and the backup takes over, the system can contain a single point of failure. If the backup resource fails before the new card is on-line, the system could potentially crash. This window of vulnerability can be minimized by keeping the period of suspension as short as possible. This requires careful planning, and gathering as much information as possible before actually suspending driver operation and powering-down a card slot.

When an extended suspension period is unavoidable, or when the system is mission-critical, it is desirable to configure a second backup resource if possible.

How to On-line Replace (OLR) a J3525A PCI Card using SMH

WARNING **When performing online replacemant, the stack for all the ports of the interface card under consideration will be stopped before repalcement. They will be restarted after the repalcement. Thus any user application which has open connections will receive DISCONNECT messages. The connections will have to be re-established after successful replacement of the interface card.**

Step 1. Start SMH.

Step 2. From the `SMH Areas` screen, select `PeripheralDevices`.

Step 3. From the `PeripheralDevices` screen, select `Cards`.

Step 4. From the `I/O Cards` screen, view the list of available I/O cards. Select from the possible entries mentioned below:

- HP J3525A PCI 2-port PSI card

Step 5. From the `Menu` bar, select `Actions`.

Step 6. From the `Actions` drop-down list, select `Replace`.

Step 7. SMH now performs a Critical Resource Analysis (CRA). That is, now that you have selected to `Replace` a card, SMH's first step is to confirm that no critical resources will be disabled when the card is taken off-line.

Output messages from the CRA process are presented in the `Analyze Critical Resources` screen which will be shown before you can proceed. The messages displayed on this screen and the availability to continue on from it ("OK" button activated) depend on the results of the analysis.

Table 4-2 Three Possible Critical Resource Analysis (CRA) Outcomes

Outcome	Notes	Screen Displays	Buttons Activated	User Actions
No critical resources identified.	At this point, you can still cancel the replacement process.	"No affected resources are critical or in-use" and "Critical Resource Analysis complete" messages.	OK and Cancel	Click "Cancel" to halt the operation and cancel the replacement with no change to the system. Or, Remove all cables from the PCI I/O card and click "OK" to take you to the next step.
Critical resource(s) identified.	SMH will not allow the operation to proceed.	Detailed message describing the affected critical resource.	Cancel	Click "Cancel" to halt the operation with no change to the system
Other resources identified.	SMH reports other resources that are in use with no detectable alternates. For these resources, you can cancel or continue the operation based on your knowledge of the current system configuration.	Detailed message describing these resources.	OK and Cancel	Click "Cancel" to halt the operation with no change to the system. Or, Remove all cables from the PCI I/O card and click "OK" to continue operations based on your knowledge of the information being reported.

NOTE The cables must be removed from the PCI I/O card before proceeding with the OLR operation.

Step 8. Once you click the "OK" button on the `Analyze Critical Resources` screen, SMH begins to take the selected card out of service. SMH requests a suspend operation for the driver of the selected card.

Step 9. Once the drivers are suspended, SMH turns off the power to the slot in which the card is located.

Step 10. SMH then illuminates the amber attention LED on the slot itself to make the suspended card more easily locatable on the system chassis.

Step 11. SMH now requests that the card be replaced via a dialog box. Read the contents of this dialog for any extra information (Expansion cabinets, warnings, etc.). Also at this point, SMH turns off the slot's green power LED.

Step 12. Replace the target card. Please refer to your system manual for removal and insertion of the PCI I/O cards.

NOTE Do not connect the cables to the card at this point.

Step 13. At this point, the amber LED should still be be activated, and the green power LED should still be off.

Return to the console, and click the "OK" button on the `Replace Card` dialog.

WARNING **Pressing the "Cancel" button here will prevent you from easily restoring power to this slot (or Power Domain). If this happens, you must shut-down, then re-boot, the system to restore power.**

Step 14. Once you click "OK", SMH first resets the attention LED to it's normal state.

Step 15. SMH completes the operation by reversing the sequence of actions. That is, SMH will:

a. return power to the card slot, and enables power to the Green LED

b. identify the new card

c. resume driver operations to the card

Step 16. Connect the cables to the card.

How to On-line Add (OLA) a J3525A PCI Card using SMH

NOTE On-line addition will recognize and claim the interface card only if the X.25/9000 software is already loaded on the system. Also check if the appropriate J3525A drivers are loaded in the kernel.

Step 1. Read the information (below) in this step. An understanding of this section is important in order for you to make the correct decision later in the procedure.

1. Enter the `SMH I/O` Cards area and look for a slot prior to inserting the card.
   - You will see one or more entries marked “empty slot”, one of which you will choose to house the new card.
   - SMH will perform a Critical Resource Analysis prior to bringing the card on-line

Step 2. Start SMH.

Step 3. From the `SMH Areas` screen, select `Peripheral Devices`.

Step 4. From the `Peripheral Devices` screen, select `Cards`.

Step 5. From the `I/O Cards` screen, view the list of available I/O slots (will read “empty slot”. Select the slot you wish to use.

Step 6. From the `Menu` bar, select `Actions`.

Step 7. From the `Actions` drop-down list, select `Add`.

Step 8. SMH now performs a Critical Resource Analysis (CRA). That is, now that you have selected to `Add` a card, SMH’s first step is to confirm that no critical resources will be disabled when the power to the slot is switched off.

Output messages from the CRA process are presented in the `Analyze Critical Resources` screen which will be shown before you can proceed. The messages displayed on this screen and the availability to continue on from it ("OK" button activated) depend on the results of the analysis.

Table 4-3 Three Possible Critical Resource Analysis (CRA) Outcomes

Outcome	Notes	Screen Displays	Buttons Activated	User Actions
No critical resources identified.	At this point, you can still cancel the replacement process.	"No affected resources are critical or in-use" and "Critical Resource Analysis complete" messages.	OK and Cancel	Click "Cancel" to halt the operation no change to the system. Or, Click "OK" to take you to the next step .
Critical resource(s) identified.	SMH will not allow the operation to proceed.	Detailed message describing the affected critical resource.	Cancel	Click "Cancel" to halt the operation with no change to the system
Other resources identified.	SMH reports other resources that are in use with no detectable alternates. For these resources, you can cancel or continue the operation based on your knowledge of the current system configuration.	Detailed message describing these resources.	OK and Cancel	Click "Cancel" to halt the operation with no change to the system. Click "OK" to continue operations based on your knowledge of the information being reported.

SMH displays a dialog indicating that the selected slot has power disabled and it is now safe to add the card to the desired slot.

Step 9. At this point, the amber LED is activated and the green LED is off. Now add the new PCI card. Refer to the system manual for the insertion of the PCI card.

NOTE Do not connect the cables to the card at this point

Step 10. Once you have inserted the new card, return to the console and click "OK" on the dialog, assuming there is one. At this point, SMH will:

a. return power to the card slot and enable power to the Green LED

b. attach drivers by running `ioscan` on the new hardware

c. update the list of cards and slots on the system shown in the I/O cards area

NOTE At this point, the OLA is complete. Note that in some cases additional configuration in another area of SMH may also be required. A network interface card, for example, might require network parameter setup in the Network Interface Card portion of the Networking and Communications area.

Step 11. If the OLA is successful, connect the cables to the card.

5 Diagnostic Utilities

Using Diagnostic Utilities

This chapter describes how to use the X.25 diagnostic utilities.

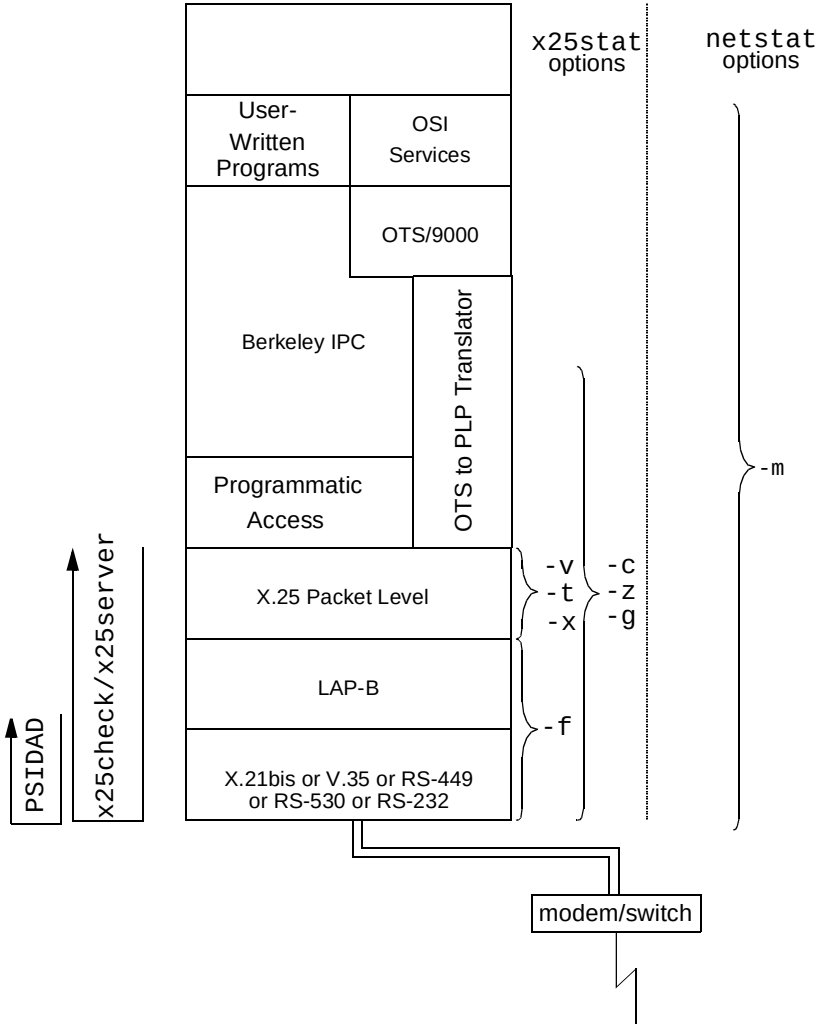
The diagnostic utilities provided with your X.25 link are described briefly in the following table. Refer to Chapter 8, Troubleshooting, if you need help deciding which utility to use.

Table 5-1 Available Diagnostic Utilities

Utility	Description	Refer to:
`x25check` `x25server`	Tests connectivity up to X.25 level 3 between the local and remote nodes.	man page and this chapter for examples
`x25stat`	Provides X.25 and IP over X.25 status and configuration VC statistics.	man page and this chapter for examples
`x25mibstat`	Uses X.25 MIB structure to provide statistical information.	man page
`netstat`	Provides network statistics and information about network connections.	man page
`ping`	IP-to-IP connections only. Tests connection to a remote host (to IP level) and reports round-trip communication time.	man page

The following diagram shows the areas of the X.25 product covered by the diagnostic utilities. The X.21bis level is physically integrated in the interface card.

Figure 5-1 Scope of X.25 Diagnostic Utilities



Before Using the Diagnostic Utilities

Some of the diagnostic utilities request or display information about the X.25 interface card using one or more of the naming conventions described below. Keep these conventions in mind when using the diagnostic utilities:

- the name of the *device file*, uses the format x25_*n*p*x*, where *n* represents the card instance number (0 to 15). For systems with multi-port cards only, the *p* is a mandatory place holder and *x* represents the port (interface) number (either 1 or 4).
- the *programmatic access name* (for example, *interfacenpx*)
- the *X.25 interface name* used by netstat uses the format x25_*n*p*x*, where *n* represents the card instance number (0 to 15). For systems with multi-port cards only, the *p* is a mandatory place holder and the *x* represents the port (interface) number (either 1 or 4).

The interrupt signal is often used to terminate diagnostic utilities. This chapter assumes you have designated the **Break** key as your interrupt character by setting the *stty* flags, *brkint* and *-ignbrk*. See the man page for stty for details.

NOTE You should be familiar with X.25 data formats in order to understand diagnostic output.

x25check and x25server

These commands are used together to test the connection between a source node and a destination node, up to and including X.25 programmatic access level. `x25check` is a major tool for debugging the X.25 subsystem and for checking the configuration for that subsystem. `x25server` is a background process that waits for call requests from `x25check`. **The `x25server` must be running before you execute `x25check`**.

See the man pages for `x25check` and `x25server` for syntax and parameter information.

NOTE The `x25server` daemon will be killed if all of the X.25 interfaces are deactivated .

Example 1: Running x25check Interactively (No Parameters)

The example below shows `x25check` used interactively to do three tests.

- The first test (sending a call packet only) is successful and the results are shown. (Generally, when running `x25check`, first send the CALL packet alone, without data, to verify that a VC has been established between the nodes).
- The second test uses the same destination address to send five data packets. Note that the X.121 address you enter becomes the default value the next time your run the test.
- The third test (to a different X.121 address) is unsuccessful. An error message stating the problem is displayed.

When `x25check` prompts you for a programmatic access name, press **Return**. The programmatic access name is not used for 712 Series workstations.

```
$ x25check
       X25CHECK (c) COPYRIGHT Hewlett-Packard Company 1988.
       Test Starts on.. Wed Dec 8 09:13:41 1993
```

```
Initialization of the test...

Do you want to send only a CALL packet (no DATA)? (y/n) > y
Enter the X.121 Address of the remote node ( ) > 2502057
Enter the programmatic access name through which you want to
check ( ) > interface2

Use reverse charging? (y/n) > y
Closed user group identification number (Blank or xx or xxxx) >

CALL packet sent ...

The following figures have been measured on the network:
           Set up time: 302 ms
Remote Connection Succeeded

Do you want to run the test once again? (y/n) > y
Initialization of the test...

Do you want to send only a CALL packet (no DATA)? (y/n) > n
Enter the X.121 Address of the remote node (250207)>
Enter the programmatic access name through which you want to check
                                                  (interface20) >

CALL packet sent ...
DATA packet sent ... DATA packet received
DATA packet sent ... DATA packet received
DATA packet sent ... DATA packet received
DATA packet sent ... DATA packet received
DATA packet sent ... DATA packet received

The following figures have been measured on the network:
          Set up time  : 311 ms
          Transit time : 120 ms
Remote Connection Succeeded

Do you want to run the test once again? (y/n) > y
Initialization of the test...

Do you want to send only a CALL packet (no DATA)? (y/n) > y
Enter the X.121 Address of the remote node ( 250207 ) > 2502059
Enter the programmatic access name through which you want to check
(interface20) > interface3
Unable to Connect to Remote Node
```

```
VC_CLEAR Packet was received with
CAUSE   0 :       DTE Originated
DIAG   245 :        Cannot Interpret Diagnostic Code
```

Example 2: Running x25check With an X.121 Address

In the example below, the destination X.121 address of the destination node is specified as a parameter to x25check. This example shows a successful test and assumes the *closed_user_group* to be 51.

```
$ x25check 250207 -i interface0 -g 51

     X25CHECK (c) COPYRIGHT Hewlett-Packard Company 1988.
     Test Starts on .. Wed Dec 8 09:15:01 1993

Initialization of the test...

CALL packet sent ...

The following figures have been measured on the network:
          Set up time : 267 ms
Remote Connection Succeeded
```

Example 3: Running x25check With a Data Packet

In this example, a DATA packet is sent following the CALL packet. The size of the DATA packet (in octets) and the number of DATA packets to be sent are specified in addition to the X.121 address of the destination node. This example shows a successful test.

```
$ x25check 250207 -i interface0 -s 32767 -n 1
     X25CHECK (c) COPYRIGHT Hewlett-Packard Company 1988.
     Test Starts on .. Wed Dec 8 09:15:47 1993

Initialization of the test...

CALL packet sent ...
DATA packet sent ... DATA packet received

The following figures have been measured on the network:
          Set up time : 286 ms
          Transit time : 24205 ms
Remote Connection Succeeded
```

x25stat

This command displays the status, configuration, and VC statistics of an X.25 interface. See the man pages on `x25stat` for syntax and parameter information.

Example 1: Displaying the Current Configuration

```
x25stat -c -d x25_0
-------- X.25 CONFIG -----------

General Parameters:
   X.121 Address: 110
   X.121 Pkt Addr: 110
   Programatic Access Name: x25interface_0
Level 1 parameters
   Linespeed: external clock
Level 2 parameters
   t1 [frame timeout]:   3000 ms
   t3 [idle timer]:      12000 ms
   n2 [retransmissions]: 3
   Max. Framesize:       269
   Level 2 Window:       7
Level 3 parameters
   network type:  DTE_8
VC parameters  ( Low  High ) values :
   PVCs      :    0    0
   SVCs In   :    0    0
   SVCs 2ways:    1   64   SVCs Out  :    0    0
   Total Number of VCs configured: 64
   Default inbound packet size:      128
   Default outbound packet size:     128
   Default inbound window size:      2
   Default outbound window size:     2
   Default inbound throughput class: 12
   Default outbound throughput class:12
Facilities settings:
   Fast Select is : DISABLED
   Flowcontrol Neg is : OFF
   Reverse Charging is : DISABLED
```

Example 2: Displaying Global Statistics

See the tables provided after this example under the headings "SUBNETWORK STATISTICS FOR X25" and "STATISTICS FOR LAPB" for help in interpreting the displayed "State" of the X.25 link.

```
x25stat -d x25_0 or
x25stat -g -d x25_0
SUBNETWORK STATISTICS FOR X25
-----------------------------
Subnetwork ID  : 0
State          : Connected and resolved DXE
-------------------------------------
Packet type              TX        RX
-------------------------------------
Restart Request           0         1
 Restart Confirm          1         0
  Calls (out/in)          1         0
   Calls Accepts          0         1
   Reset Request          0         0
   Reset Confirm          0         0
 Packets (total)         30        29
   Bytes (total)       3840      3712
-------------------
STATISTICS FOR LAPB
Subnetwork  : 0
Link mode   : LC_LAPBDTE
Link state  : NORMAL
------------------------------------------------------------
  FRAMES      TX_CMD      TX_RSP      RX_CMD      RX_RSP
------------------------------------------------------------
Supervisory:
  RR              22          27           0          23
RNR                0           0           0           0
REJ                0           0           0           0
Unumbered:
  SABM             0                       1
  DISC             0                       0
DM                 0                       0
UA                 1                       0
FRMR               0                       0
Information:
  I               33                      62
------------------------------------------------------------
                  TX                                  RX
------------------------------------------------------------
```

```
Other:
 Bad length                                          0
 Unknown                                             0
 Erroneous                                           0
 Discarded                                           0
 Ignored           0                                 0
 Retransmitted     0
------------------------------------------------------------
Timers:
 T1               0
 T4               0
 T4 (N2 times)    0
STATISTICS FOR WAN
------------------
Subnetwork Link State : HDLC_ESTB
        WAN:
              83  good frames transmitted
              87  good frames received
               0  transmit underruns
               0  receive overruns
               0  CRC/frame errors received
               0  received frames with no buffer
                   0  received frames with no flow control
                   0  receive buffer overflows
```

The level 3 “State” under the heading “SUBNETWORK STATISTICS FOR X25” can be:

Table 5-2 Level 3 State

Output on Screen	Description/Meaning
Link not up	X.25 link level 3 is down
Connecting to DXE [a]	Restart packet sent (r2 [b])
Connected resolving DXE	Restart packet sent (r2)
Random wait started	Waiting before retransmitting a restart after a “collision” (packet sent at the same time as packet received)
Connected and resolved DXE	X.25 link level 3 normal (r1)
DTE RESTART REQUEST	Waiting on restart confirmation

Table 5-2 Level 3 State (Continued)

`Waiting link disc reply`	Level 2 going down
`Buffer to enter WtgRES`	Restart sent after retry
`Buffer to enter L3restarting`	Waiting for restart reply during "link going down" phase
`Buffer to enter L_disconnect`	Waiting for buffer to clear VCs during "link going down" phase
`Unknown`	Unable to define error/problem

a. DXE is DCE or DTE
b. r2, like r1, is a CCITT definition of a state/condition

The level 2 "Link state" under the heading "`STATISTICS FOR LAPB`" can be:

Table 5-3 Level 2 Link State

Output on Screen	Description/Meaning
`NORMAL`	Level 2 up
`ADM`	DTE - Level 1 disconnected
`POLLING`	DCE - Level 1 disconnected
`START`	Level 1 connected
`RESET`	Level 2 down
`OFF`	Level 2 down

NOTE T4 under "`STASTISTICS FOR LAPB`" is only an internal parameter; it is not configurable.

The level 1 "`Link state`" under the "`STASTISTICS FOR WAN`" heading can be:

Table 5-4 Level 1 Link State

Output on Screen	Description/Meaning
`HDLC_ESTB`	WAN connection established
`HDLC_IDLE`	No WAN connection established
`HDLC_DISABLED`	WAN connection disabled

The output under the heading "`STATISTICS FOR WAN`" is explained in the table below:

Table 5-5 Statistics for WAN Output

Output on Screen	Description/Meaning
`transmit underruns`	Shows the number of times a transmission was aborted because the next octets to be transmitted were not provided soon enough for the level 2 firmware
`receive overruns`	Shows often-received octets were overwritten because they were not processed by the card
`received frames with no buffer`	No message available in level 1 to copy received data to
`received frames with no flow control`	Never used
`receive buffer overflows`	Number of frames that were "too long"
`receive aborts`	Number of frames that have been aborted

Example 3: Displaying Virtual Circuit Data Packet Counters

The following example shows the output when there is one virtual circuit connected.

Note that the virtual circuit specified in the command line is 64, in decimal. The LCN shown in the example output is displayed as 40, the hexadecimal equivalent of 64.

```
x25stat -t 64 -d x25_0

PER-VC STATISTICS FOR X25
-------------------------
Subnetwork     : 0
LCN            : 040 (hexadecimal)
User ID        : 213
Call direction : outward To DTE         : 202
To DTE         : 202
VC state       : 6 - Datatransfer
-------------------------------------
Packet type              TX        RX
-------------------------------------
             Call         1         0
     Call confirm         0         1
             Data       422       421
        Interrupt         0         0
              RNR         0         0
               RR         0       421
            Reset         0         0
    Reset confirm         0         0
            Clear         0         0
    Clear confirm         0         0
-------------------------------------
Total                   423       843
-------------------------------------
```

The "VC state" under the heading "PER-VC STATISTICS FOR X25" can be:

Table 5-6 VC State

Output on Screen	Description/Meaning
3 - P2	CALL REQUEST sent, but not yet replied

Table 5-6 VC State (Continued)

4 - P3	INCOMING CALL received, but not yet replied
6 - Datatransfer	Connection established
7 - DXE[a] busy	RNR sent
8 - D2	RESET REQUEST sent, but not yet replied
19 - DXE resetting	RESET INDICATION received, but not yet replied
20 - P6	CLEAR REQUEST sent, but not yet replied

a. DXE is DCE or DTE.

Example 4: Displaying Global X.25 Level 3 Statistics

```
x25stat -x -d x25_0

SUBNETWORK STATISTICS FOR X25
-----------------------------
Subnetwork ID  : 0
State          : Connected and resolved DXE
-------------------------------------
Packet type              TX        RX
-------------------------------------
Restart Request           0          1
Restart Confirm           1          0
Calls (out/in)            3          0
Calls Accepts             0          3
Reset Request             0          0
Reset Confirm             0          0
Packets (total)           1273       1271
Bytes (total)             162944     162688
```

Example 5: Displaying X.25 Level 1 and 2 Statistics

```
x25stat -f -d x25_0

STATISTICS FOR LAPB
------------------
```

```
Subnetwork  : 0
Link mode   : LC_LAPBDTE
Link state  : NORMAL
-----------------------------------------------------------
  FRAMES      TX_CMD      TX_RSP      RX_CMD      RX_RSP
-----------------------------------------------------------
Supervisory:
RR              45        1045           0          46
RNR              0           0           0           0
REJ              0           0           0           0
Unumbered:
SABM             0                       1
DISC             0                       0
DM                           0                       0
UA                           1                       0
FRMR                         0                       0
Information:
  I             1280                    2551
-----------------------------------------------------------
                  TX                                 RX
-----------------------------------------------------------
Other:
Bad length                                           0
Unknown                                              0
Erroneous                                            0
Discarded                                            0
Ignored           0                                  0
Retransmitted     0
-----------------------------------------------------------
Timers:
 T1               0
 T4               0
 T4 (N2 times)    0
STATISTICS FOR WAN
------------------
Subnetwork  : 0
Link State : HDLC_ESTB
WAN:
                2371  good frames transmitted
                2599  good frames received
                   0  transmit underruns
                   0  receive overruns
                   0  CRC/frame errors received
                   0  received frames with no buffer
                   0  received frames with no flow control
```

```
0  receive buffer overflows
0  receive aborts
```

NOTE T4 under "STASTISTICS FOR LAPB" in the example above is only an internal parameter; it is not configurable.

The output under the heading "STATISTICS FOR WAN" is explained as follows:

Table 5-7 Statistics for WAN Output

Output on Screen	Description/Meaning
`transmit underruns`	Shows the number of times a transmission was aborted because the next octets to be transmitted were not provided soon enough for the level 2 firmware
`receive overruns`	Shows often-received octets were overwritten because they were not processed by the card
`received frames with no buffer`	No message available in level 1 to copy received data to
`received frames with no flow control`	Never used
`receive buffer overflows`	Count of "too long" frames received
`receive aborts`	Count of frames that have been aborted

Example 6: Displaying Virtual Circuit Status

```
x25stat -v -d x25_0

PER-VC STATISTICS FOR X25
-----------------------------
```

```
LCI  Type     VC State     Subnetwork Local address Remote Address
---  ----     --------     ---------- ------------- --------------
040  SVC-2way Datatransfer 0          110            202
```

Note that the LCI number given in this example output is displayed in hexadecimal, not decimal.

Example 7: Displaying Current IP to X.25 Address Mapping

x25stat -a

```
IP Address  PktszWindow VCs CUG   SUB-REV  BAR-REV  X25 Address
192.25.0.12     1       1   -     y        n        0.250107..
192.25.0.22     1       1   GOO11 y        n        0.250112..
192.25.0.32     1       1   -     y        n        0.250208..
```

Some of the fields in the example above are:

- CUG – An example Call User Group would be *GOO11*, which means group ll.
- SUB-REV – Subscribe to Reverse charges (y=yes, n=no).
- BAR-REV – Stop Reverse charges (y=stop, n=don't stop).

x25mibstat

This is a programmatic application that can be used to display statistics for X.25 sub-systems. Refer to the `x25mibstat(1)` man pages for instructions on using this utility.

Some examples of `x25mibstat` statistics are provided in the following table.

Table 5-8 Example Statistics for x25mibstat

Statistic	x25mibstat Option	Remark
VC open duration	`-s x25CircuitEntry` `-s mioxPeerEntry`	Time duration in seconds Time duration of IP connection
Number of open SVCs per X.25 connection	`-s x25StatEntry`	According to VC type (that is, inbound, outbound, or 2-way)
Clear information	`-s x25statEntry` `-s mioxPleEntry`	About clear initiator About refused calls
Clear information	`-s lapbFlowEntry`	Number of LAPD-B state changes

6 PAD Services

Introduction

This section describes X.25/9000 PAD (Packet Assembler/Disassembler) Services which enable HP 9000 computers to connect and communicate with remote systems over a Packet Switching Network (PSN).

NOTE Refer to "Configuring PAD Services" in Chapter 3 for information on configuring PAD Services with SMH.

PAD Services include the following commands and functions:

- x29server – for PAD support for remote terminals:

  A daemon process that allows a local HP 9000 system to function as a host for (real or emulated) terminals attached to a remote PAD.

- x29printd – for remote PAD printer support:

  A daemon process that allows local HP 9000 computers to print at printers attached to a remote PAD.

- x29uucpd – for PAD-UUCP connectivity:

  A daemon process that allows HP 9000 computers to originate UNIX-to-UNIX copy services over an X.25 Switched Virtual Circuit (SVC).

- padem – for local PAD emulation:

  A user command that allows a local HP 9000 system to emulate both a terminal and a PAD for connection with remote systems over a PSN.

CCITT Recommendations

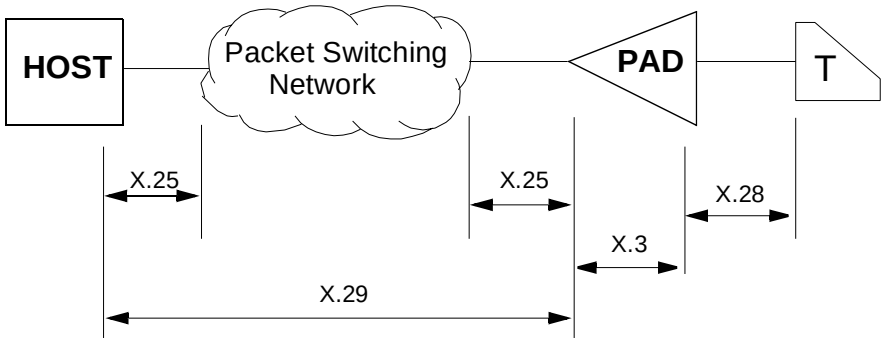
X.25/9000 PAD Services are compliant with the X.25, X.28, X.29, and X.3 CCITT Recommendations of 1984.

Recommendation X.28 defines the exchange of messages between PAD terminals and the PAD. A superset of the commands defined by this protocol are performed by `padem`.

Recommendation X.29 is the protocol for the exchange of messages between the host and PADs.

Recommendation X.3 defines a set of parameters that control PAD operation. The following illustration shows the relationship between each of these recommendations.

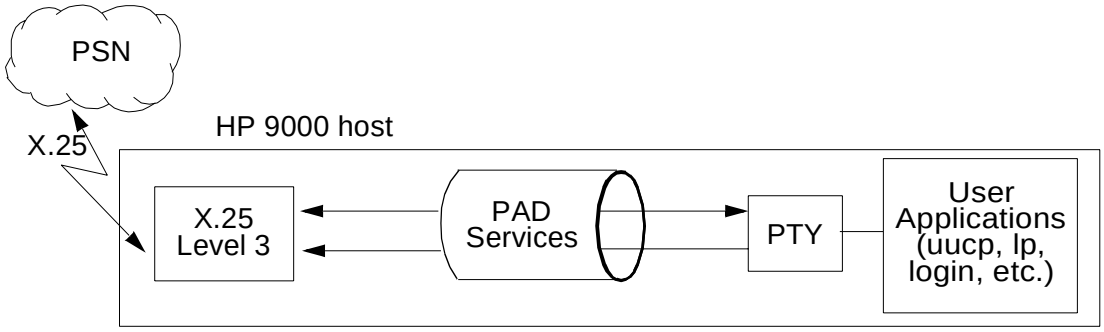
Figure 6-1 PAD Support Protocols



PAD services and the HP 9000 host

All of the PAD services can be seen as a pipe through which data is passed from X.25 to the PTY (Pseudo Type) terminal and vice versa (see the illustration below). Terminal data is not modified, but the X.25 headers are stripped before data reaches the PTY.

Figure 6-2 PAD Services as a Pipe

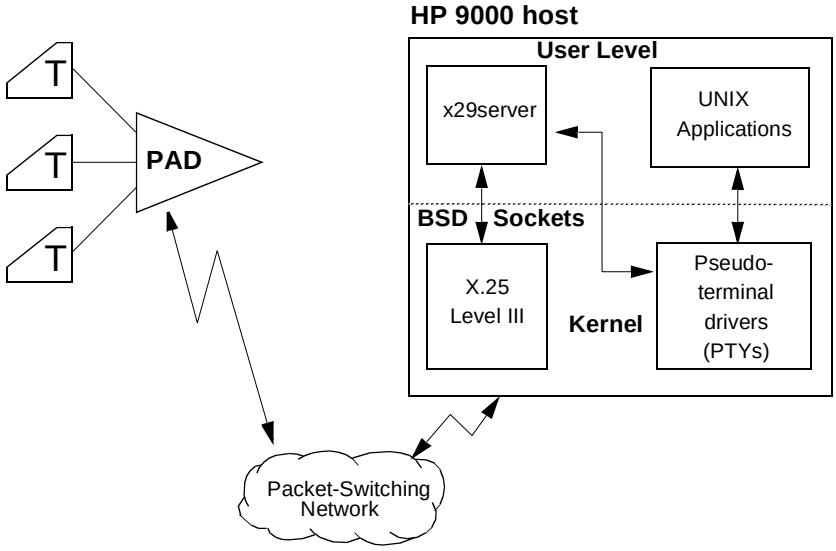


Remote PAD Support (x29server)

Overview

The `x29server` user-level process provides support for communications with terminals attached to a remote PAD. This process accesses X.25 level 3 through Berkeley Sockets (programmatic interface) and has an interface to PTYs (see illustration below).

Figure 6-3 Remote PAD Support



The x29server remote PAD support provides capabilities for:

- Managing call acceptance.
- Launching applications automatically.
- Ensuring system security.
- Monitoring access and data transfers.

When x29server begins execution, it examines the PAD support section (*pad_spt*) of the /etc/x25/x29hosts file. The PAD support section contains user-configurable parameters for all devices. See Appendix B, X.25 Configuration Files and Examples, for an example of the x29hosts file.

Call Acceptance Mechanism and System Security

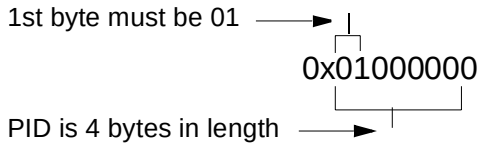
The x29server process operates in conjunction with devices declared in the set of *pad_spt* (PAD support) entries contained in the x29hosts file. It continually "listens" to the network and initializes communications channels at the user level when a call request is received. x29server can handle a large number of calls because each communication channel is managed by an independent process.

The x29server process employs Berkeley Sockets (at X.25 level 3) to set up dedicated listening sockets that only handle calls arriving at a particular subsystem. Addressing information contained in incoming call request packets enables the listening process to determine if the call should be accepted.

When a call request packet arrives x29server looks in the *pad_spt* entries to verify the following:

- **Legal PID in the incoming call request packet** – The PID must conform to the X.29 standard to ensure support on all X.25 networks. x29server only verifies that the PID begins with 01 and that it is 4 bytes in length as illustrated below.

Figure 6-4



- **Correct interface_name** – If there is an entry in the *interface_name* field in the *pad_spt* section (optional), it must contain the name of the interface over which the call has arrived.
- **Correct address** – If there is an entry in the *local_x121* field in the *pad_spt* section (optional), this entry must match the called address specified in the call request packet. Furthermore, the *local_x121* field entry must contain the complete address (that is, *both* the address and subaddress of the local interface).

  If all of the above conditions are met, the call is connected and x29server does the following:

  — parses the x29hosts file to find the *first* previously accepted *pad_spt* entry whose *remote_x121* field matches the calling address contained in the call request packet,

  — and, if there is a *pty_slave_fname* field in the selected *pad_spt* entry, x29server verifies that the corresponding PTY master/slave pair exists and is available.

If the last two items are completed successfully, the call is accepted; otherwise, x29server rejects the call.

NOTE See the man pages for `x29server` for its syntax and parameters.

Supported Remote PAD Terminals

- 700/41, 700/92, 700/94, 700/22
- 2392A
- 2393A
- Vectra with AdvanceLink Terminal Emulation

Configuring Remote PAD Support

You can configure remote PAD support with SMH or by editing configurable parameters in the `/etc/x25/x29hosts` and `/etc/x25/x3config` files. The `x29hosts` file contains general parameters for each remote device and the `x3config` file contains X.3 communications profile parameters for each device declared in the `x29hosts` file. See Appendix B, X.25 Configuration Files and Examples, for examples of the `x29hosts` and `x3config` files.

Configuring pad_spt Parameters

Remote PAD support parameters are specified in the *pad_spt* section of the `x29hosts` file. An example *pad_spt* section with parameter values is shown below.

```
pad_spt {

   interface_name   interface0
   remote_x121      408555120801
   local_x121       1235451
   pty_slave_fname  ptynb01
   application      /bin/login2
   cud              yes
   logging          1
   reverse_charge   disable
   size_parity      8_none
   x3               hp_padsrvr
}
```

pad_spt Parameters

This section describes the remote PAD support parameters shown in the above example:

interface_name **Optional**. This is a character string (maximum 12 characters) with no wildcards allowed. If this field is not present, `x29server` will listen on all X.25 interfaces.

remote_x121 **Mandatory** (no default value). This is the calling address (the address of the remote system).

There are three ways of expressing addresses:

- exact address: the complete, exact address
- wildcard address: partially or completely composed of legal wildcards
- special address: for PADs with no local address

Legal characters include the digits 0 through 9, the capital letter "F", the question mark (?), and the asterisk (*). The (?) is a legal wildcard (substitute) for values 0 through 9. The (*) is a legal wildcard for any character. You can mix wildcards and characters.

The special address is FFFFFFFFFF which maps to a null calling address. This feature is supported for PADs that have no local addresses.

local_x121 **Optional**. Specified with a maximum of 16 digits (no wildcards allowed). This must be the complete address (including *both* the subscription address of the interface and a subaddress).

The complete address protects the interface from being activated by calls arriving at other addresses.

pty_slave _fname **Optional**. Specifies the file containing the name of the slave PTY driver that will provide terminal services for user applications. The name of this file (maximum 14 characters) must begin with "`pty`" (format: *`pty [filename]`*) and the file must be located in the `/dev/pty/` directory.

The name of the corresponding master PTY driver must also exist in a file beginning with "ptym" (format: *ptym [filename]*) located in the /dev/ptym/ directory:

Wildcards are not allowed in these file and path names.

If the *pty_slave_fname* field contains an entry, x29server considers any communication initiated by this entry to be restricted to exclusive use by the TTY/PTY (master/slave) pair. The call is refused if the TTY/PTY pair does not exist, is already in use, or cannot be accessed (permission should be set to 666).

logging **Optional**. Specifies the logging level for each call. Log information is stored in the file /var/x25/log/x29server/x29log*XXXXX*, where *XXXXX* is the process ID of the current x29server child process. Logging can be at levels 0, 1, 2, or 3. Level 0 is no logging and the default is level 1 (error logging). Level 2 is error and warning logging. Level 3 is error, warning, information, and status logging.

reverse_charge **Optional**. Specifies reverse charge options for the calling address. Possible settings are: *disabled* (or *disable*), *enabled* (or *enable*). With the default value (*disable*), reverse charge calls will not be accepted. When reverse charging is enabled and is requested by the remote PAD or system, the call is accepted and is charged to the local system. You must also configure the X.25 interface card to allow reverse charging (refer to "Configuring Remote System Access" in Chapter 3 for instructions on how to use SMH to enable reverse charging) in order for reverse charging to work with PAD services.

size_parity **Optional**. Legal values are *8_none* (the default), *7_even*, and *7_odd*. Specifies the character size and parity setting of the remote PAD and terminal. This parameter is useful when the asynchronous port of the PAD cannot be configured in 8 bits. In this case, this parameter must be set to match the remote PAD and terminal settings. x29server will set the PTY pair correctly so that 7-bit PAD users can log in without

problems. Note that size and parity processing will only be performed on input from the PAD to the system.

x3 — **Optional**. This is the X.3 configuration set name for this `x29server` session. If this entry is included in the `/etc/x25/x29hosts` file, the configuration name and its set of X.3 parameters must also be specified in the `/etc/x25/x3config` file. If more than one X.3 configuration set with the same name exists in `x3config`, the first one is used. X.3 parameters are not downloaded unless *both* the X.3 configuration set name *and* the parameters are given.

application — **Optional**. Used with *cud* to specify the full path name of the application to be launched when the connection is established (maximum 80 characters, no wildcards). See "Launching applications automatically" below for more information.

cud — **Optional**. Specifies a **yes** or **no** condition. Legal values for this field are: *enable* (or *yes*), *disable* (or *no*). See "Launching applications automatically" below for more information on this.

Launching Applications Automatically

A user-level application can be launched automatically by configuring the *application* and *cud* fields in the *pad_spt* section of the `x29hosts` file.

The *application* parameter specifies the name of the user application (or script) to be automatically launched when the connection is established. The *cud* parameter is used like an on/off switch to enable or disable the use of Call User Data (CUD) instructions.

If you are using a UNIX script, it must contain the character string *#!/bin/ksh* at the beginning of the first line. If this string is not included, the system will return an error code.

If the *application* and *cud* fields do not appear in the *pad_spt* file, the `/bin/login` (default) program is automatically launched.

CAUTION

Use extreme caution when specifying applications other than the default (*login*) application. Since the `x29server` process is started with *superuser* privileges, it gives root privileges to applications triggered by incoming calls.

You can automatically launch a user application by:

- configuring the *application* field with the relevant application name as its value without specifying the *cud* field. The application name must be the full path name of the application to be launched.
- configuring the *application* field with the application name and setting the *cud* field to `yes`. In this case, the *cud* field is considered as the argument to the application. The application name must be the full path name of the application to be launched.
- not including the *application* field and setting the *cud* field to `yes`. This allows the application to be defined by the incoming Call Request Packet.

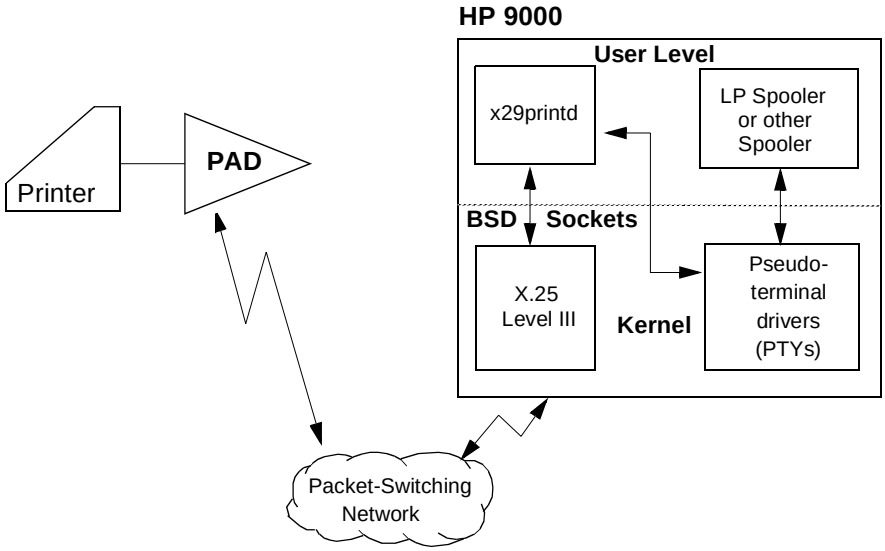
The first four bytes of the CUD field (in call request packets) are *NOT* part of the application's full path name. The first four bytes are used for protocol ID as explained in the section "Call acceptance mechanism and system security."

Remote PAD Printer Support (x29printd)

Overview

The `x29printd` user-level program provides support for printers attached to remote PADs. It accesses X.25 level 3 through Berkeley Sockets (programmatic interface). It also has an interface to special PTYs (see illustration below).

Figure 6-5 Remote Printer Support



When `x29printd` begins execution, it examines the `/etc/x25/x29hosts` file for all PAD devices configured with a PAD type printer. For each of these configured devices, `x29printd` monitors all print requests sent to the spooling system (see Appendix B, X.25 Configuration Files and Examples, for an example of the `x29hosts` file).

When `x29printd` detects a print request, it establishes an SVC with the remote PAD printer specified in the remote X.121 address. `x29printd` then transmits the print data to the remote PAD printer, closes the SVC, and waits for another print request.

System Requirements

Because each printing device requires system resources, HP recommends that the kernel parameters be set to following values (or greater):

- *MAX_PAD_PR* – is the number of PAD printers to be configured.
- *NPTY* – is 60 + *MAX_PAD_PR*
- *MAXURPC* – is 50 + (3 x *MAX_PAD_PR*)
- *NPROC* – is 20 + ((8 x *MAXUSERS*) + *NGCSP* + (6 x *MAX_PAD_PR*))

The following kernel parameters should not be modified and should retain the values indicated below:

- *NFILE* =(16 x (*NPROC* + 16 + *MAXUSERS*)/(10 + 32+(2 x *NPTY*))
- *NINODE* = ((*NPROC* + 16 + *MAXUSERS*) + 32 + (2 x *NPTY*) + (*SERVER_NODE* x 18 x *NUM_CODES*))

For more information on dependencies, refer to your *System Administration Tasks* manual.

NOTE See the man pages for `x29printd` for its syntax and parameters.

Configuring Remote PAD Printers

You must configure each remote PAD printer by specifying parameters in the printer section of the `/etc/x25/x29hosts` file (see Appendix B, X.25 Configuration Files and Examples, for an example `x29hosts` file). An example printer section is shown below.

```
printer {
      device          printer1
      name            interface0
      remote_x121     408555111201
      cud             dbad
      cug             02
      logging         1
      reverse_charge enable
      x3              hp_printer
}
```

device **Mandatory**. This is the name of the device file that will be located in `/dev/x29/`. It is created by `x29printd` and will be symbolically linked to a slave PTY in `/dev/pty/`. The device file must also be configured in the spooler system using `lpadmin` and the *-v* option (see "Configuring the UNIX Line-printer Spooler for x29printd" below).

name **Mandatory**. This is the name of the programmatic interface that will be used for call setup. There is no default for this entry. This name must match the programmatic access name (also called *name*) specified in the `x25init` X.25 configuration file.

remote_x121 **Mandatory** (no default). This is the X.121 address of the PAD device plus the subaddress of the printer. This address is required for calls to be initiated. Only digits 0 through 9 are legal.

cud **Optional**. This is the Call User Data (CUD) appended to the protocol ID (0x01000000) for any call request packets sent to this printer. Only strings of 12 characters or less are permitted.

cug **Optional**. This is the Call User Group (CUG) that will be allowed access to the printer. Only integers are permitted.

logging **Optional** (default = 1). This is the logging level used for each printer device. Printer device logging can be set to 0, 1, 2, or 3 where the respective values are:

0 – No logging (do not create a log file).

1 – Minimum session error logging.

2 – Error and warning logging.

3 – Error, warning, information, and status logging.

HP recommends logging level 0 or 1 for normal operation. Logging levels 2 and 3 should only be used for troubleshooting for short periods of time.

Logging messages for `x29printd` provide descriptions of what the messages mean and the actions required to correct any errors that may occur. The log file is written to

`/var/x25/log/x29printd/[`*device_name*`][`*child_process_id*`]`.

reverse_charge **Optional** (default = *disabled*). Specifies the reverse charge option for the call setup for the particular printer address. It can be *disabled* (or *disable*), *enabled* (or *enable*).

x3 **Optional**. This is the X.3 configuration set name for this `x29server` login session. If used, the configuration set name (and X.3 parameter set) must also exist in `/etc/x25/x3config`. If more than one X.3 configuration set with the same name exists, the first one found in the file is used.

If there is no configuration set name in `/etc/x25/x29hosts` or no matching X.3 configuration set name in `/etc/x25/x3config`, default values are used.

Configuring the UNIX Line-printer Spooler for x29printd

The remote PAD printer must also be configured in the spooler system in order for remote printing to work with the standard UNIX LP commands. To configure a remote PAD printer in the spooler system, execute the following command:

```
lpadmin -v <device name> -p <printer name> -m <printer model>
```

-p Printer name is the name given to this printer at the user level. This is the same name specified with the `lp` command's *-d* option (used when you print a file to a printer).

-v Device file name is `/dev/x29/[`*device name*`]`. The value for device name is the same character string as the one entered in the device entry in the `/etc/x25/x29hosts` file.

-m Printer model is the HP printer model name (for example, *thinkjet*, *laserjet*, *hp2563a*, etc.).

The `lpadmin` command associates the printer name with the device file name in `/etc/x25/x29hosts` and maps this name to the remote X.121 address of the printer.

Examples

x29printd and lpsched Operation

For this example, a printer (a RuggedWriter) is connected on port 5 (or B1 on HP 2335a PADs) to a PAD whose address is 4085551203. The address of the printer in this case would be 408555120305. In the PAD device local configuration, port 5 must be downloaded with:

```
parm_no: parm_value;1:0; 2:0; 3:0; 4:10; 5:0; 6:0; 7:0;
8:0; 9:0; 10:0; 11:14; 12:1; 13:0; 14:0; 15:0; 16:8;
17:24; 18:0; 19:1; 20:0; 21:0; 22:0
```

which is profile 21 on HP 2335a.

On the host side, the printer name is *foo*. This means that when a user types **`lp -d foo /tmp/file`**, the file `/tmp/file` will be printed on this printer. The X.25 interface on the host is **`interface0`**.

Printer Configuration

To configure a printer (foo in the examples below) for use with the standard UNIX LP spooler system and `x29printd`, follow these steps:

Step 1. Edit the `/etc/x25/x29hosts` file to include:

```
printer {
        device          x29printer1
        name            interface0
        remote_x121     408555120305
        x3              default_printer
        logging         3
}
```

`x29printd` uses `/etc/x25/x29hosts` to configure remote PAD printers.

Step 2. Execute `/usr/sbin/x29printd` with the *`-l3`* option.

After `/usr/sbin/x29printd` starts, it creates a device file named `/dev/x29/x29printer1`. You can verify that this device file exists with the command: `ll /dev/x29`.

`x29printd` creates the device file by linking it to a slave PTY. This master/slave PTY pair is the interface for `lpsched` and `x29printd` (see the pty(7) man pages for information about UNIX PTYs). The device file for printer *foo will be written to* `/dev/x29/x29printer1`.

Step 3. Configure the printer in the spooler system with the following commands:

```
lpshut
lpadmin -v/dev/x29/x29printer1 -pfoo -mruggedwriter
lpsched
accept foo
enable foo
```

`lpshut(1M)` is necessary since `lpadmin` will not execute when `lpsched` is running. You must restart LP scheduler after `lpadmin`. `lpsched(1M)` schedules requests taken by `lp(1)` for line printing. After a printer is configured in the spooler, you must enable it to accept printing requests.

Verifying the Configuration

After you complete the steps above, check the configuration as shown below:

- `lpstat -t` should display the following (assuming that only this printer is configured in the spooler system and you did these steps on January 23, 11:00 am):

```
scheduler is running
no system default destination
device for foo: /dev/x29/x29printer1
foo accepting requests since Jan 23 11:00
printer foo is idle.  enabled since Jan 23 11:00
fence priority : 0
```

- `ps -ef` should show that both `lpsched` and `x29printd` are running.
- `/var/x25/log/x29printd/x29printd.log` should have been created and should contain information on the configuration of `/dev/x29/x29printer1`.
- `/var/x25/log/x29printd/x29printd.log` should not contain any configuration error messages.

UUCP Support (x29uucpd)

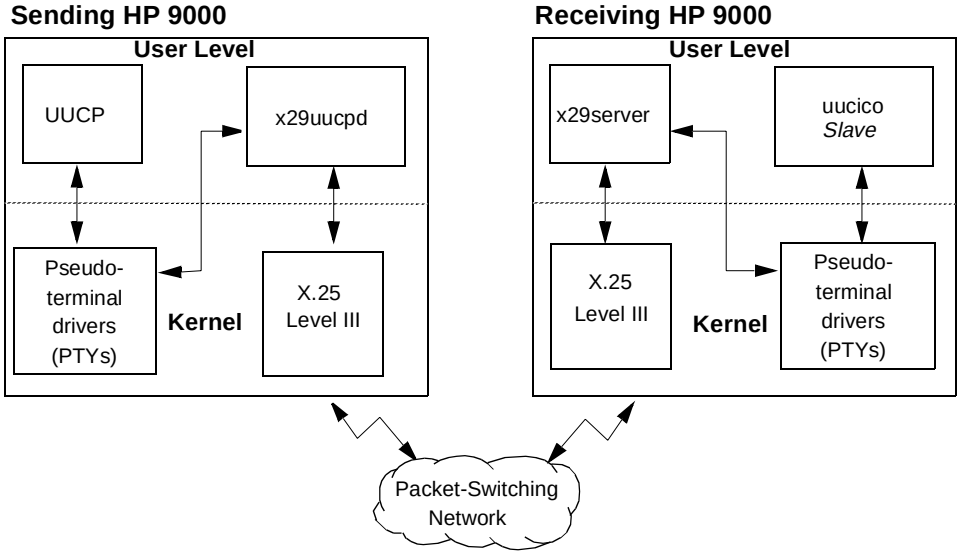
Overview

x29uucpd provides UUCP connectivity on X.25 networks using CCITT Recommendations X.3 and X.29. With x29uucpd, users on HP 9000 host systems with X.25 access and UUCP can execute UUCP subsystem commands to other systems running X.25, PAD support, and UUCP. x29uucpd interfaces with the X.25 subsystem through BSD Sockets to initiate call request packets to remote systems on X.25. When a call request arrives at a destination system, it is received by the PAD support service.

NOTE x29uucpd creates device files in the /dev/x29/ directory. The device files must be configured in the UUCP subsystem in order for UUCP to interface with x29uucpd.

The x29uucpd user-level program provides support for UUCP file transfers. It accesses X.25 level 3 through Berkeley Sockets (programmatic interface) and uses PTY pairs for the terminal interface to UUCP (see the following illustration).

Figure 6-6 Remote Printer Support



When `x29uucpd` is executed, it examines the `/etc/x25/x29hosts` file for all PAD devices that are configured for UUCP.

When `x29uucpd` receives a copy request, it establishes an SVC with the remote host specified by the X.121 address and transfers the data.

If the remote host is an HP 9000, the `x29server` process monitors that system for incoming calls. `x29server` and `x29uucpd` cooperate to handle all SVC requirements while the UUCP processes on each system perform the UUCP operations.

When the UUCP data transfer is complete, the UUCP processes terminate. `x29server` and `x29uucpd` then terminate the SVC connection.

Configuring UUCP PAD Support

To configure PAD support for UUCP services, the `x29hosts` file and other system and device files must contain the appropriate references and parameters. This section provides information on configuring these files. See Appendix B, X.25 Configuration Files and Examples, for examples of the `x29hosts` and `x3config` files.

NOTE See the man pages for `x29uucpd` for full details on syntax and parameters.

Table 6-1

File	Contents
`/etc/x25/x29hosts pad_uucp`	parameters for each remote host
`/etc/passwd`	uucp entry
`/etc/x25/x3config`	optional X.3 parameters
`/usr/lib/uucp/Systems`	remote system's name
`/usr/lib/uucp/Permissions`	remote system's login name
`/usr/lib/uucp/Devices`	device file name

Configuring the x29hosts file

The `/etc/x25/x29hosts` file must contain a separate *pad_uucp* section for each remote site. An example *pad_uucp* section is shown below.

```
pad_uucp {
      device          x25uucp
      name            interface0
      remote_x121     4085551113
      cud             abc
      cug             2
      logging         3
      reverse_charge  enable
      x3              hp_uucp
}
```

pad_uucp Parameters

device **Mandatory**. This is the device file name located in the `/dev/x29` directory. It is created when `x29uucpd` is executed and is used for communication between UUCP and `x29uucpd`. This device file must also exist in `/usr/lib/uucp/Devices` file for the UUCP system.

name **Mandatory**. This is the name of the programmatic interface that will be used for call setup. There is no default for this entry. If this entry does not exist, the call setup will not be completed and file transfer will not take place. This name must match with the programmatic access name (also called name) specified in `x25init`.

remote_x121 **Mandatory**. This is the X.121 address of the X.25 interface card of a host system where the PAD support program (for example, `/usr/sbin/x29server`) is running. This address is required for calls to be initiated. There is no default for this address.

cud **Optional**. This is the Call User Data (CUD) appended to the protocol ID (0x01000000) for any call request packets sent to this device. Strings of 12 characters or less are permitted.

cug **Optional**. This is a the Call User Group (CUG) number (a positive value between 0 and 99) embedded in the call request packet sent to this device.

logging **Optional** (default = 1). Specifies the logging level for each UUCP connection to this device. The log file is `/var/x25/log/x29uucpd/[device_name][child_pid]`. Logging levels for this UUCP device are 0, 1, 2, or 3 where:

0 – No logging (no log file is created).

1 – Error logging.

2 – Error and warning logging.

3 – Error, warning, information, and status logging for troubleshooting.

HP recommends using log level 0 (no logging) for normal operation.

reverse_charge **Optional** (default is *`disable`*). Specifies the reverse charge option for call setup at this address. Possible settings are *`disabled`* (or *`disable`*) and *`enabled`* (or *`enable`*). When reverse charge is enabled, `x29uucpd` sends out call request packets with reverse charge requested.

x3 — **Optional**. This is the X.3 configuration set name for UUCP file transfers. If this entry is used in `/etc/x25/x29hosts`, the configuration set name with its set of X.3 parameters must be specified in `/etc/x25/x3config`. All legal X.3 parameters contained in the `x3config` file are downloaded at connection time. If there is more than one X.3 configuration set with the same set name in `/etc/x25/x3config`, the first one is used.

If there is no configuration set name in `x29hosts` or matching X.3 configuration set name in `x3config`, the default parameters are downloaded (see "X.3 Parameter Descriptions" for information on X.3 parameters). `x29uucpd` operates in line mode only.

Configuring System and Device Files for UUCP

The following example describes the configuration procedure for `cu`, `uucp` and `uucico` transfers between two systems. In this example, the local system is called "*dave*" and the remote system is called "*bill*." The two systems are configured to handle file transfers originating from the local system (*dave*) and received by the remote system (*bill*).

NOTE Refer to the man pages for information on `x25uucp` device entry parameters.

At the "dave" system:

Step 1. Make sure that the directory `/var/spool/uucp/LCK..x29` exists. If it doesn't exist, create it.

Step 2. Configure the `usr/lib/uucp/Systems` file as follows:

```
bill Any;5 dev_bill, f 19200 - "" \r\d\r\d\r login: -BREAK
-login: uucp word: test
```

where:

`bill` – is the name of the destination system (7 characters maximum).

Any – indicates that *bill* can be called at any time.

5 – is the retry duration (*dave* will try again to connect to *bill* in 5 minutes).

dev_bill – is the name of the remote system declared in *dave*'s Devices file.

f – is the file transfer protocol. HP recommends using *f* rather than *g* for better performance.

19200 – is the transfer rate (it must be set at the same speed as that specified in the *dave*'s `Devices` file).

"" \r\d\r\d\r login: -BREAK -login: uucp word: test – is login information declaring **uucp** as the user for `uucp` and `uucico` processes, with the password "**test**". The password creates a link to the entry in the `/etc/passwd` file of *bill*.

Step 3. Make sure that the following entry appears in the `/usr/lib/uucp/Devices` file:

```
dev_bill x29/bill - 19200 direct
```

x29/bill has a maximum of 14 characters. The **x29/bill** entry creates a link with the **pad_uucp** entry in the `x29hosts` file.

Step 4. Add the following lines to the `/usr/lib/uucp/Permissions` file:

```
MACHINE=bill \

SENDFILES=yes REQUEST=yes CALLBACK=no \

READ=/

WRITE=/

COMMANDS=ALL
```

NOTE You can verify current permissions with the `uucheck -v` command.

Step 5. Create the following entry in the `/etc/x25/x29hosts` file:

```
pad_uucp {
       device       bill
       name         interface0
       remote_x121  [bill's X.121_address]
       logging      3
       x3           default_x3
}
```

Refer to "Configuring the x29hosts file" earlier in this section for the definitions of these parameters.

Step 6. Launch `/usr/sbin/x29uucpd` at the HP-UX command prompt.

At the "bill" system:

Step 7. Make the following modifications to the `pad_spt` section of `/etc/x25/x29hosts` file:

```
pad_spt {

    remote_x121  [dave's X.121_address or
                  "*" for all addresses]
    logging      3
    x3           hp_padsrvr
}
```

Refer to "Configuring the x29hosts file" earlier in this section for the definitions of these parameters.

Step 8. Launch the `x29server` process with **`/usr/sbin/x29server -l3`** for example.

Step 9. Verify the contents of `/etc/passwd` (this file is linked with *dave's* `Systems` file) which should look like this:

```
uucp::5:3::/var/spool/uucppublic:/usr/lib/uucp/uucico
```

At the terminal prompt, assign the password "test" to the `uucp` user:

```
bill# passwd uucp
```

Step 10. Add the following lines to the `/usr/lib/uucp/Permissions` file:

```
LOGNAME=uucp \

SENDFILES=yes REQUEST=yes CALLBACKI=no \
```

**READ=/ **

WRITE=/ \COMMANDS=ALL

where the **LOGNAME** parameter creates a link with *dave's* `Systems` file.

Step 11. Test the X.25 connection using:

bill# x25server
dave# x25check [*bill's X.121 address*]

If successful, you will receive the message "`Remote connection succeeded.`"

Step 12. Test the files: `x29hosts`, `x3config` (at *bill*) using:
dave# padem *[bill's X.121 address]*

If successful, you will receive the "`login`" message and you will be able to enter your details.

Step 13. Test the files: `x29hosts`, `x3config`, `Systems`, `Devices` (at *dave*) using:
dave# cu -d9 bill

If successful, you will receive an output something like:

```
Autodialing - please wait
call dial(1761259328)
baud=19200, speed=19200, line=(null), telno=-
call find_dev(15076527500)
baud=19200, speed=19200, line=(null), telno=-
nomodem is 0
fixline(6, 19200)
fixline - direct
MDTR was not set
gdial(direct) called
call mode(1)
Connected
transmit started
receive started
Generic-Sys (generic) [HP Release A.B9.04]

login:
```

If you don't see the "`login:`" prompt, type **dave# uusnap**. You should receive the following output:

```
x29   --- --- --- LOCKED (pid 57602)
```

```
bill  --- --- --- [error message]
```

Refer to the UUCP section in Chapter 9 of the *Remote Access: User's Guide* (part number B2355-90037) for more information on UUCP error messages.

Step 14. Test the file permissions of both *bill* and *dave* using:

```
dave# /usr/lib/uucp/uucico -r1 -x9 -sbill
```

where **-r1** indicates that *dave* is the "master" (for UUCP purposes) and **-x9** is the highest debugging level.

If successful, you should receive output similar to:

```
mchFind called (bill)
list (rmail) num = 1
list (/) num = 1
list (/) num = 1
list (ALL) num = 1
_Request (TRUE), _Switch (TRUE), _CallBack (FALSE), _MyName
(dave),
_Commands ALL
chdir(/var/spool/uucp/bill)
conn(bill)
ProtoStr = f
Device Type dev_bill wanted
mlock x29/bill succeeded
fixline(5, 19200)
fixline - direct
gdial(direct) called
              /* X.25 connection established*/
getto ret 6
expect: (“”)
got it
sendthem (^MDELAY
^MDELAY
^M^M)
expect: (ogin:)
             /* reception of “bill”’s banner*/
^MGeneric-Sys (generic) [HP Release A.B9.04]^J^M^J^Mlogin:got it
sendthem (uucp^M)
enter ub_sst, status is : 0
Rmtname: bill
imsg > ^M^M^Jlogin: ^M^M^Jlogin: ^M^M^M^Jlogin: ^M^Jlogin:
uucp^M^M^JPlease wait
..checking for disk quotas^M^J^PShere=bill^@Login Successful: System=bill
```

```
imsg >^PROK^@msg-ROK
Rmtname bill, Role MASTER,  Ifn - 6, Loginuser - root
rmesg - 'P' imsg >^PPgfdx^@got Pgfdx
wmesg 'U'f
Proto started f
               /*the f protocol for UUCP is selected*/
*** TOP ***  -  role=1, setline - X
gtwvec: dir /var/spool/uucp/bill
wmesg 'H'
got HY- 'H' enter frdmsg:HY
PROCESS: msg - HY
HUP:
wmesg 'H'Y
cntrl - 0
send OO 0,imsg >HY^M^POOOOOO^@exit code 0
Conversation Complete: Status SUCCEEDED
```

If you receive an error such as "`RETRY TIME NOT REACHED`," you can delete the status file as follows:

```
dave# rm /var/spool/uucp/.Status/bill
```

If you receive the message "`SUCCEEDED`," your uucp/X.25 connection is operating correctly.

Step 15. Test file transfers using:

```
dave# uucp -r /tmp/filename bill!/tmp/filename
```

where you're transferring a file from *dave* to *bill* and the `-r` option puts the request in the queue.

To launch the file transfer correctly, now use:

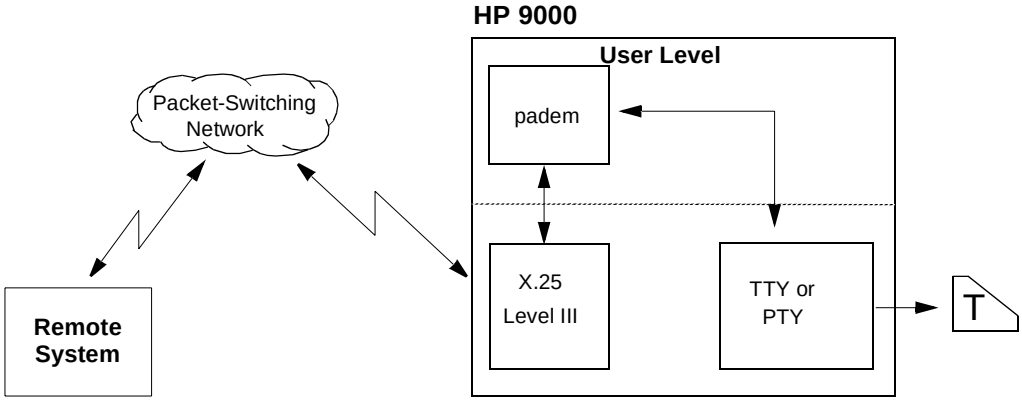
```
dave# /usr/lib/uucp/uucico -r1 -x9 -sbill
```

This completes the testing procedure.

Local PAD Emulation (padem)

The padem program provides local PAD emulation to enable a local terminal (real or emulated) to connect to a remote system over an X.25 PSN. x29server handles the connection on the remote host side. The remote system can be specified upon execution or with subsequent padem commands issued during operation (see illustration below).

Figure 6-7



The padem program operates in either command or data transfer mode. The operating mode depends on whether an X.121 address or a symbolic address is specified when padem is launched.

Command Mode

If the X.121 address or symbolic address of the remote host is not specified, padem begins execution in command mode and prompts the user for commands with X28> (the padem prompt).

In command mode, padem receives X.28 PAD commands from users. Any padem command can be entered at the prompt. Once padem establishes an SVC with a remote host, it switches to data transfer mode.

Data Transfer Mode

If the X.121 address or the symbolic address of the remote host is specified, padem begins execution in data transfer mode.

In data transfer mode, typed text is sent to the remote host and text received from the remote host is displayed at the local terminal. You can re-enter command mode to issue a padem command by pressing **Ctrl P**.

If the VC is cleared at any time (by the remote host or by the X.25 network), padem returns to command mode or exits if the remote host address was at the HP-UX prompt.

Configuring Local PAD Emulation

NOTE See the man pages for padem for its syntax and parameters.

You can configure local PAD emulation by modifying parameters in the *pad_em* and *host_table* sections of the /etc/x25/x29hosts file. Examples of these sections are given below. See Appendix B, X.25 Configuration Files and Examples, for an example of the x29hosts file.

```
pad_em {
       remote_x121       4085551111
       name              interface0
       logging           3
       reverse_charge    enable
       profile           0
}
host_table {
       Gale              4085551111
       Tornado           4085551113
       Deepthought       4085551115
}
```

Parameter Descriptions

remote_x121 **Mandatory**. This is the X.121 address of the remote system. Only numbers 0 through 9 are legal; wildcards or special address types are not allowed. If this entry exists, any call to the remote X.121 address will use the associated fields (*logging*, *reverse_charge*, *profile*, and *name*).

name	**Optional**. This is the name of the programmatic interface used for call setup. If this entry does not exist, the default name associated with */dev/x25_0* is used. This name must match the programmatic access name (also called *name*) in the X.25 configuration file.
logging	**Optional**. Specifies the logging level that applies to all calls made during this session. The log file is `$HOME/plog.aXXXXX`, where the last five characters are from the process ID of `padem`. If this parameter is not specified, the logging level is set to the default level of 0 (no logging). Logging levels are: 1 = error logging, 2 = error and warning logging, 3 = error, warning, information, and status logging.
reverse_charge	**Optional**. Specifies the reverse charge option for call setup. Possible settings are: *disabled* (or *disable*) and *enabled* (or *enable*). The default value is *disabled*. When reverse charge is enabled, `padem` sends a reverse charge request with all call request packets.
profile	**Optional**. Describes the X.3 configuration for the login session. The X.3 default values are defined in the code (programmatically) and cannot be altered by users. These default parameters comply with the 1984 *profile 0* specification for a PAD device. If *profile X* is in `x29hosts` and *profile Y=X* is in `x3config`, `padem` uses *profile X*. If there is no profile in `x29hosts`, and `x3config` contains *profile 0*, `padem` uses *profile 0*. In all other cases `padem` uses the hard-coded default values. `padem` only operates in line mode for X.3 configuration.
host_table{}	**Optional**. Defines symbolic host names for remote X.121 addresses. The first entry defines the symbolic name and the second entry is the remote X.121 address. If an X.121 address is given, it must also exist in the *remote_x121* field in the *pad_em* section of the `x29hosts` file.

NOTE

You do NOT need to have *pad_em {}* entries in the `x29hosts` file. You can specify all of the above parameters at the command line when you run `padem`.

Command line options are given priority over those in the `x29hosts` file. If options are not specified at the command line or in the `x29hosts` file, the programmatic access name associated with `/dev/x25_0` is used for call setup, the logging level is set to 1, the reverse charge option is *`disabled`*, and the default X.3 profile values are used.

PAD Commands

PAD command signals are grouped into the X.28 PAD command set (defined by CCITT X.28 recommendations) and include an extended command set providing additional functionality.

X.28 PAD Command Set

clr

Clears a virtual circuit (VC), disconnecting the VC from the remote host.

int

Transmits an interrupt to the remote host.

par?

Displays all of the X.3 parameters and their current values. The format for the display is:

```
X.3 parameter number: current value
```

prof [profile_id]

Interprets a prepared set of command signals from a profile. Profiles must be placed in the file `/etc/x25/x3config`.

reset

Resets a virtual circuit (VC). When a VC is reset, all data received (but unread) is lost, and all X.3 parameters are set to the default values (as they were defined before the VC was established).

set n:v [,n:v]

Sets the X.3 parameter *n* to value *v*. If there is a syntax error, `padem` responds with an ERR service signal and updates the diagnostic text. All changes take effect when a new VC is established, or immediately if one is already established.

set? n:v [,n:v]

Sets the X.3 parameter *n* to value *v* and displays the current setting of all parameters. This command is a combination of the `set` and `par?` commands.

stat

Displays status information of a VC. `padem` responds with "`FREE`" if no VC is established or with "`ENGAGED`" if there is a VC established.

[r,g[n[n]]-] x121_address [subaddress] [P/Dd..d

Performs a virtual call to the remote host defined by its X.121 address. *x121_address[subaddress]* can be replaced with "**`|symad|`**" where **`symad`** is defined in the file `/etc/x25/x29hosts`. **`P`** specifies the packet size negotiation. **`D`** specifies the Call User Data (CUD).

Extended Command Set

aprof Lists all profile IDs known to `padem`. The profile IDs are contained in the `x3config` file.

help Displays a complete list of all `padem` commands with a short description of each.

ifname [string]

Displays or sets the programmatic access name of the interface to which `padem` is (or will be) connected.

If this command is entered with a programmatic access name specified in *`string`*, `padem` sets the programmatic access name for the interface. If the command is entered and *`string`* is not specified, `padem` displays the programmatic access name for the interface.

list Lists all symbolic names for remote hosts that are accessible to the user. The symbolic reference names are contained in the `/etc/x25/x29hosts` file.

lprof Displays the last loaded profile ID.

quit Terminates `padem`. If the SVC is still connected, it is cleared.

sleep s Suspends `padem` for *s* seconds (maximum 3600).

verbose	Puts padem in verbose display mode. When padem operates in verbose display mode, it displays CAUSE, EFFECT, and ACTION messages in addition to the standard diagnostic message.

NOTE

Certain parameters may be changed automatically during the session as a result of PTY reconfigurations by the application at the other side of the connection. These changes will temporarily alter the original defaults set by the PAD Services. The PAD terminal user may also change PAD parameters at any time with X.28 commands. However, this may result in unpredictable behavior from the application being processed.

Configuring X.3 Profile Parameters

X.3 PAD parameters are used to specify how certain I/O operations are performed between Data Terminal Equipment (DTE) and the remote host PAD. These parameters are used to ensure that the local DTE and the remote PAD use the same communications profile.

Each of the 22 parameters has an identifier known as the parameter reference. This identifier is an integer value from 1 to 22. Each parameter has a defined range of possible values as specified by the X.3 recommendations.

X.3 parameters are specified in one or more configuration sets contained in the `/etc/x25/x3config` file. Each configuration set includes:

- X.3 configuration set name and profile number
- Parameter number
- Parameter value for line mode (value1) and raw mode (value2)
- Comment lines starting with "#"

See Appendix B, X.25 Configuration Files and Examples, for an example of the `x3config` file.

Configuration Set Syntax

The syntax for an X.3 configuration set is shown in the following example:

```
config_name : profile_number {
      # <X.3 parameter_no>   <value1>   <value2>
      1                      1          1
      2                      1          1
      .                      .          .
      .                      .          .
      .                      .          .
      22                     0          0
}
```

An X.3 configuration set is a string which embodies the group of X.3 parameters and parameter values beginning with "{" and ending with "}". Each configuration set is identified by a configuration set name (*config_name*) and an optional profile number (*profile_number*).

Parameter Values

config_name **Mandatory**. Specifies the configuration set name, a character string that identifies the X.3 parameter set. Each *config_name* must be unique. This is a required parameter for all PAD services if user-defined X.3 parameters are to be used. This name must match the *x3* value given in the x29hosts file for *pad_spt*, *printer*, or *pad_uucp* entries. Since X3 has no effect on **pad_em** entries, a dummy string for *config_name* together with a profile number should be entered for padem profile parameters.

profile_number **Optional**. Applies only to the padem configuration. This is a number that must match the number entered for *profile* in the **pad_em {}** section of the x29hosts file. If the profile entry exists and is assigned a value in the x29hosts file, it must also appear in the x3config file.

X.3_parameter _no **Mandatory**. This is an integer value from 1 to 22. See "X.3 parameter descriptions" later in this chapter for a full description of each X.3 parameter.

value1, value2 **Mandatory**. These are the values to be used in X.3 configuration. Each value is defined for a specific X.3 parameter: *value1* is for line-mode configuration; *value2* is for raw-mode configuration. Only x29server implements raw-mode data transfer and only parameters 3 and 4 are used for raw-mode data transfer. x29printd, x29uucpd, and padem do not use *value2*.

Modifying Parameters

Initial parameter settings can be changed at any time during a call. This occurs when x29server detects an ioctl() call to configure the PTY for a PAD terminal. This system call could be issued by the UNIX login process, by a shell initialization script such as .login, or by an application.

Each ioctl() request is examined by the server. If a mapping exists between the terminal I/O parameter and X.3, the terminal I/O value is checked against the current X.3 setting. If the request calls for a change,

the current X.3 setting is overwritten on the server. A set command is sent to reset the parameter in the remote PAD. If `x29server` cannot map the terminal I/O parameter to X.3, the request is processed as a normal terminal I/O `ioctl()`.

When `x29printd` and `x29server` transform terminal I/O parameters into X.3 parameters, they intercept the `ioctl` (`tcseta`) call and send an X.29 set command with new X.3 parameter values if necessary. The related X.3 parameters are: 2 (echo), 3 (data forward), 4 (idle timer), 5 (device control), 12 (flow control), 15 (editing), 16 (character delete), and 17 (line delete).

The `x29server` process downloads only those X.3 parameters that are specified in the `x3config` file for a particular remote system. The `x29print` process always downloads a set of X.3 parameters (if they exist in the `x3config` file) at the beginning of a connection, or the default X.3 parameters (given in the preceding table) if the X.3 parameters do not exist in the `x3config` file.

The X.3 parameters for `padem` can be changed by an X.29 set message from the remote host or by the terminal operator with a set command.

Default X.3 Parameters

If the `x3config` file does not have a definition for a particular device, the defaults (listed in the next section) are used. See Appendix B, X.25 Configuration Files and Examples, for an example of the `x3config` file.

X.3 Parameter Descriptions

The following pages contain a description of X.3 parameters 1 through 22. The default X.3 parameters vary for each PAD service.

NOTE Parameters 1 to 18 are available for networks using either the 1980 or 1984 standard. Parameters 19 through 22 only apply to the 1984 standard.

Table 6-2 X.3 Parameter Descriptions

X.3 Parameter	Description
1 - ESC/Data Transfer	Specifies whether or not the terminal user will be allowed to escape from data transfer mode in order to send PAD command signals. 0 indicates escape is not allowed, 1 permits escape from data transfer mode with **Ctrl-P**, 32 to 126 permits escape from data transfer mode with the defined character.
2 - Echo	Specifies whether or not the PAD echoes to the local terminal all the characters received from that device. 1 turns echoing on; 0 turns echoing off.
3 - Data Fwd Signal 1	Specifies one of the defined sets of characters for use as a data forwarding signal. When one of the characters in the set is received from the terminal, the PAD terminates the assembly of a packet and forwards it to the remote host. The possible values are listed below.

Table 6-3

Value	Meaning
0	No data forwarding character.
1	Alphanumerics.
2	Carriage Return only. (This is the default.)
4	ESC, BEL, ENQ, and ACK only.
8	DEL, CAN, and DC2 only.

Table 6-3 (Continued)

16	ETX and EOT only.
32	HT, LT, VT, FF only.
64	NUL, SOH, STX, BS, LF, CR, SO, SI, DLE, DC1, DC3, DC4, NAK, SYN, ETB, EM, SUB, FS, GS, RS, and US (all other characters in the first two columns of the International Alphabet #5).
126	All characters, except alphanumerics.
127	All characters are data forwarding.

NOTE The values listed for parameter 3 may be combined by specifying the sum of the values. For example, specifying 10 is the same as specifying 2 and 8.

Table 6-4

X.3 Parameter	Description
4 - Idle Timer	Specifies a timer duration interval in units of 1/20 of a second. If the time between characters entered at the terminal exceeds the specified timer value, the packet being assembled by the PAD is automatically transmitted. The range is 0 through 255. If 0 is specified, there is no idle timer and packets will never be transmitted due to a time-out.
5 - Anc Device Control	The ancillary device control parameter permits flow control toward the terminal by the PAD. The PAD indicates its readiness to receive characters from the terminal by transmitting X-ON (DC1) or X-OFF (DC3) to the terminal. 0 indicates that flow control by the PAD is not operational, and 1 permits flow control by the PAD.

Table 6-4 (Continued)

6 - Service Signals	Specifies that the terminal is to receive the PAD service signals that are received by the PAD. PAD service signals are described in *X.25: The PSN Connection*. 0 indicates that no PAD service signals are received by the terminal, and 1 indicates that the terminal will receive PAD service signals.
7 - On Break	Specifies the action to be taken by the PAD when it receives a break signal from the terminal. The possible values are shown below.

Table 6-5

Value	Meaning
1	Transmit interrupt packet.
2	Reset.
4	Transmit indication of break (PAD message)
8	Escape from data transfer mode.
16	Discard output to terminal.

NOTE The values listed for parameter 7 may be combined by specifying the sum of the values.

Table 6-6

X.3 Parameter	Description

Table 6-6 (Continued)

8 - Discard Output	This parameter restores data delivery to the terminal after the reception of the break character and value 16 has been specified for X.3 Parameter 7. This value may be modified by the reception of the break character. 0 indicates normal data delivery, and 1 indicates that data is discarded.
9 - Number of CR Padding Chars	Specifies the number of buffer characters to be added following a carriage return. This is used for hard copy terminals to allow time for the carriage to return to the left side of the page. The range is 0 through 7.
10 - Chars per Line/Wrap -Around	Specifies the number of printing characters to be transmitted before the PAD inserts a carriage return in the data stream. This parameter is not needed on terminals with auto-wrap-around. The range is 0 through 255.
11 - Baud	This parameter specifies the terminal access speed (Baud rate). `x29server` does not allow users to change the Baud rate. It defaults to the terminal Baud rate, which is usually 9600. The possible values and their associated Baud rates are listed below.

Table 6-7

Value	Baud Rate
0	110
1	134.5
2	300
3	1200
4	600
5	75

Table 6-7 (Continued)

6	150
7	1800
8	200
9	100
10	50
11	75 from terminal, 1200 to terminal
12	2400
13	4800
14	9600
15	19200
16	48000
17	56000
18	64000

Table 6-8

X.3 Parameter	Description
12 - Flow Control	Specifies whether the terminal can control the flow of data from the PAD to the terminal. 0 indicates no flow control by the terminal, and 1 indicates that flow control by the terminal may be used.
13 - Line Feed	Specifies whether the PAD inserts a line feed when a carriage return is detected. The possible values are shown below.

Table 6-9

Value	Meaning

Table 6-9 (Continued)

1	Line feed inserted after carriage return in data stream to the terminal.
2	Line feed inserted after carriage return in data stream from the terminal.
4	Line feed inserted after carriage return when echoed to the terminal.

NOTE The values listed for parameter 13 may be combined by specifying the sum of the values. For example, specifying 3 is the same as specifying 1 and 2.

Table 6-10

X.3 Parameter	Description
14 - Line Feed Padding	Specifies the number of padding characters to be added following a line feed. This is used for hard copy terminals to allow time for the paper to be moved up. The range is 0 through 7.
15 - Editing	Specifies whether data editing is allowed from the terminal during data transfer (for example, character delete and line delete). 1 indicates that editing is allowed; 0 indicates that no editing is allowed.
16 - Character Delete	When editing is allowed, specifies which character to use for character delete. The range is 0 through 127.
17 - Line Delete	Specifies which character to use for the line delete character (when editing is allowed). The range is 0 through 127.

Table 6-10 (Continued)

18 - Line Display	Specifies which character to use for the line display character (when editing is allowed). The range is 0 through 127.
19 - Edit Service Signals	Specifies whether and how PAD service signals can be edited. This parameter is valid when the setting of X.3 Parameter 6 is not 0. The possible values are shown below.

Table 6-11

Value	Meaning
0	No editing permitted.
1	Editing permitted for printing terminals.
2	Editing permitted for display terminals.
8 OR 32 to 126	Editing permitted using specified character.

NOTE If you specify a value greater than 7, that value specifies the editing character. This parameter is available only for networks using the 1984 standard.

Table 6-12

X.3 Parameter	Description
20 - Echo Mask	This parameter specifies which characters will not be echoed. X.3 Parameter 2 - Echo takes precedence over this parameter. The possible values are shown below.

Table 6-13

Value	Meaning

Table 6-13 (Continued)

1	Do not echo carriage return.
2	Do not echo line feed.
4	Do not echo VT, HT, and FF.
8	Do not echo BEL and BS
16	Do not echo ESC and ENQ.
32	Do not echo ACK, NAK, STX, SOH, EOT, ETB, and ETX.
64	Do not echo characters specified in X.3 Parameters 16, 17, and 18.
128	Do not echo NUL, SO, SI, DLE, DC1, DC4, SYN, CAN, EM, SUB, FS, GS, RS, US, and DEL.

NOTE The values above may be combined by specifying the sum of the values. For example, specifying 3 is the same as specifying 1 and 2. This parameter is available only for networks using the 1984 or 1988 standards.

Table 6-14

X.3 Parameter	Description
21 - Parity	Specifies how parity should be treated. The possible values are shown below.

Table 6-15

Value	Meaning
0	No parity checking or generation.
1	Parity checking.
2	Parity generation.

Table 6-15 (Continued)

3	Parity checking or generation.

NOTE This parameter is available only for networks using the 1984 or 1988 standards.

Table 6-16

X.3 Parameter	Description
22 - Number of Chars Before Page Wait	Indicates whether an end of page wait should occur and, if so, the number of lines per page. The range is 0 through 255. This parameter is available only for networks using the 1984 standard.

7 Tracing and Logging Utilities

Introduction

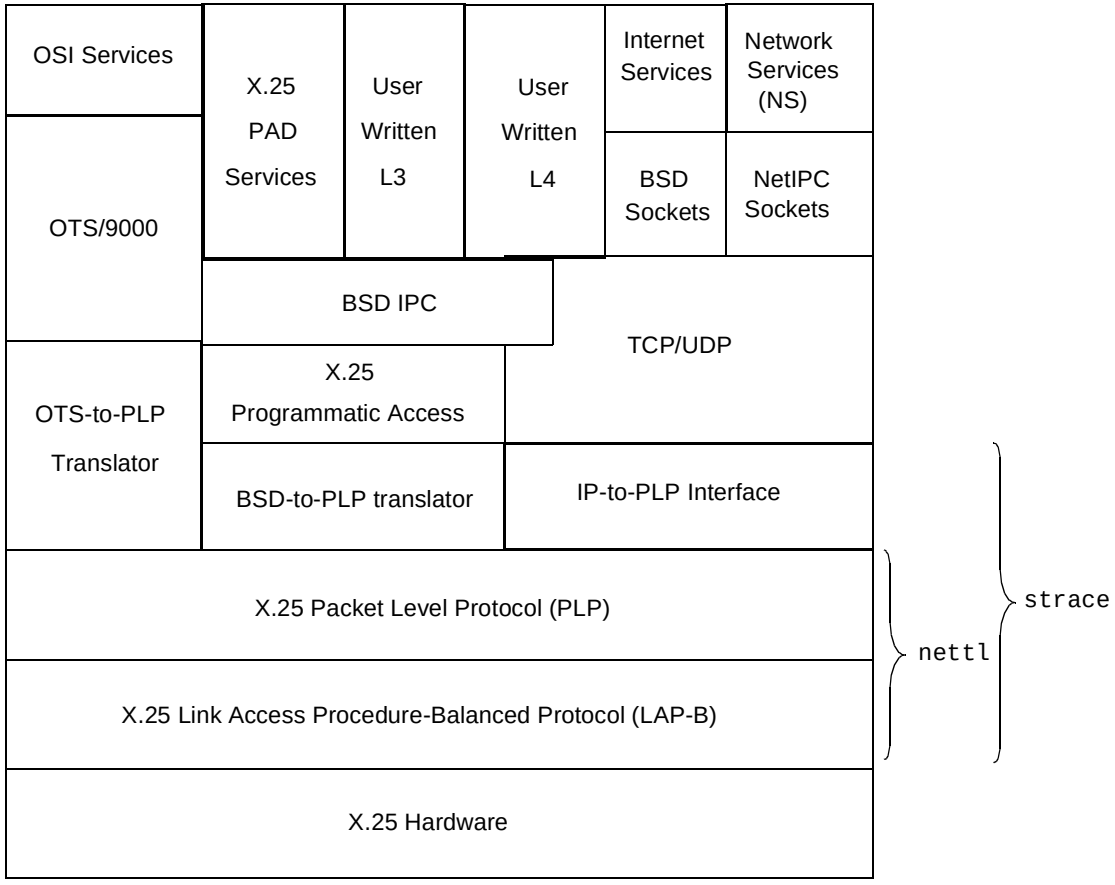
This chapter describes the use of the X.25 tracing and logging utilities provided with your X.25 link. These utilities are briefly described in the following table. Refer to Chapter 8, Troubleshooting, if you need help deciding which utility to use.

Table 7-1 Available Tracing and Logging Utilities

Utility	Description	Refer to:
`nettl`	Creates a user-defined log file containing frame and packet trace information from X.25 packet and LAP-B levels.	man page and this chapter for examples
`netfmt`	Formats trace information contained in the user-defined log file that was generated with the `nettl` command	man page and this chapter for examples
`strace`	Creates a log file of event messages for X.25 levels 2 and 3, and IP over X.25.	man page and this chapter for examples
`strerr`	Receives error messages from the Streams log driver.	man page and this chapter for examples

The following diagram shows the areas of the X.25 product covered by the network event logging and network tracing utilities.

Figure 7-1 The Scope of X.25 Tracing and Logging Utilities



nettl

Use this command to create an X.25 trace file. The description below is limited to the context of X.25 and assumes that `nettl` is already operating on your system. For more information and examples for this command, refer to your man pages.

NOTE Before you can use the `nettl` command, your X.25 connection must be operating (launched with the `x25init` command).

Syntax

```
nettl [-traceon kind [kind...]][-entity subsystem
[subsystem...]] [-file name] [-card dev name] [-size limit]
[-tracemax maxsize] [-m bytes]

nettl [-traceoff] [-entity subsystem [subsystem...]]
nettl [-status info]
```

Parameters

-traceon(-tn) Starts tracing on the specified subsystem(s). The *traceon* argument must be accompanied by the *-entity* and *-card* options. The *kind* (parameter) mask must also be specified. All operations on specified subsystems are recorded when there is a match to the *kind* mask.

-tn kind

The *kind* parameter defines the masks (keywords with associated mask values) used by the tracing facility before recording messages.

You can use any of the following keywords and mask values:

Table 7-2 Tracing Masks

keyword [a]	mask
hdrin	0x80000000
hdrout	0x40000000
pduin	0x20000000
pduout	0x10000000

a. Use *hdrin* or *hdrout* when you only want packet header information displayed. Use *pduin* or *pduout* when you want both the packet header AND the data displayed.

You can specify masks separately or combine them into a single number. For example, to enable tracing for both *pduin* and *pduout* (all packets coming in and out of the node) use the value *0x30000000*.

-entity (-e) Enter **-e SX25L2** for level 2 tracing or **-e SX25L3** for level 3 tracing.

-file (-f) Use this option to specify an output file for tracing information (mandatory unless tracing has already been started on another interface). Use the *-traceon* option the first time you run `nettl` to specify an output file.

NOTE The `.TRCX` suffix is automatically appended to the output file where `X` starts at 0 and changes to 1 when the file becomes full. Traces are always written to the `.TRC0` file.

-card(-c) This option is required with the `traceon` command to specify the X.25 subsystem interface (port) name. The X.25 device is specified with `x25_`*npx* which identifies the interface defined in your X.25 configuration file.

NOTE

The device file must be called x25_*n*p*x*, where *n* represents the mandatory card instance number (range 0 to 255). The card instance number is the number reported by the `ioscan -f` command output (in the "I" column). The (lower case) p is a place holder and *x* represents the port number (1 or 2). p and *x* are required only for systems with dual-port cards.

-traceoff (-tf) This option turns tracing, specified with the *-entity* option, off.

-status(-ss) trace Use this option to get a report on the tracing/logging status.

Examples

To turn tracing on at Level 3 for device *x25_1* (interface number 1) and receive packet header AND packet data information:

```
nettl -tn pduin pduout -e SX25L3 -c x25_1 -f /tmp/tracex25
```

To turn tracing off at Level 3 for the device *x25_1*:

```
nettl -tf -e SX25L3 -c x25_1
```

netfmt

Use this command to format the trace file that was created with `nettl`. Refer to the man pages for `x25trace` for more information and examples on this command

Syntax

`netfmt [-c` *`filter file`* `[-p]] [-F] [-t` *`records`*`] [-N] [-v] [-l] [-1[LT]] [[-f]` *`file_name`*`]`

Parameters

`-c filter file`

Specifies the name of the file containing the filter configuration commands. You create this file with your text editor. If this option is omitted, filter configuration commands are read from the `$HOME/.nettr` file if it exists. Refer to "Creating a filter file" at the end of this section for an explanation of filter file syntax.

`-p`

The parse input option lets you do a syntax check on the configuration file specified with the *`-c`* option. If the syntax is correct, `netfmt` terminates with no output or warnings.

`-F`

The follow input file option causes `netfmt` to keep the input file open when it reaches the end of the file. The file is kept open and `netfmt` continues to read from it as new data arrives. This option is useful for troubleshooting because it lets you monitor events in real time.

`-t records`

Lets you specify how many records you want to format from the end of the file. This enables you to get the most recent information. The maximum number of records is 1000. By default, all records are formatted.

`-f file_name`

Use this option to specify the binary file containing the log or trace data. This is the name of the `.TRC0` file that was specified when `nettl` was run the first time with the *-traceon* and *-file* options. By default, data is read from standard input.

-v Enables "verbose" mode, but also gives the full data output for each packet header (see *Examples* section below).

-N Enables "Nice" formatting of output. Only packet headers and the first few lines of the main data output are displayed. Only the data length is provided in addition to the packet header to make the output easier to read (see *Examples* section below).

-1 (one) Enables "terse" (short) mode formatting of each traced packet on a single line. Output lines will be more than 80 characters long if the *-L* and/or *-T* options are used (see *Examples* section below).

-l (small letter "l") Use this option when you send formatted trace data to a line printer to turn off inverse video highlighting of all trace fields.

Examples

To format the file `/tmp/tracex25.TRC0` in "verbose" mode (packet headers plus full data output for each header) with no filtering:

```
/etc/netfmt -v -f /tmp/tracex25.TRC0 | more
```

The output will resemble the following:

```
vvvvvvvvvvvvvvvvvvvvvvvLAN/X.25NETWORKINGvvvvvvvvvvvvvvvvvvvvvvvvvvv
Timestamp : Wed Sep 28 1994 10:16:19.696042
Process ID           : 10          Subsystem         : SX25L3
User ID ( UID )      : 0           Trace Kind        : 0x10000000
Device ID            : 0           Path ID           : -1
Connection ID        : 0
Location             : 00123
 Tx board: 0 at Wed Sep 28 1994 10:16:19.689181 Data Packet lci 64
    Q-bit : 0         D-bit : 0       M-bit : 0
    P(R)  : 0         P(S)  : 0
    User Data =  20 bytes
```

```
 0: 00 00 00 01 61 61 61 61 61 61 61 61 61 61 61 61 ...aaaaaaaaaaaa
16: 61 61 61 61 -- -- -- -- -- -- -- -- -- -- -- -- aaaa..........
```

To format the file `/tmp/tracex25.TRC0` in "Nice" mode (output is mainly packet headers) with no filtering:

`/etc/netfmt -N -f /tmp/tracex25.TRC0 | more`

The output will be:

```
Tx  board: 0  at Wed Sep 28 1994 10:16:19.689181 Data Packet lci 64
      Q-bit : 0         D-bit : 0       M-bit : 0
      P(R)  : 0         P(S)  : 0
      User Data =  20 bytes
```

To format the file `/tmp/tracex25.TRC0` in "terse" (short) mode, with no filtering:

`/etc/netfmt -1 -f /tmp/tracex25.TRC0 | more`

The output will be:

```
Tx  board: 0  at 10:16:19.689181 Data Packet lci 64  PR: 0 PS: 0
```

Creating a Filter File

You can use a filter file to filter information according to various criteria. This enables you to apply system-level filtering to get information such as a time stamp for a specific subsystem.

You create a filter file with your text editor by entering the filter configuration commands in predetermined fields on the same line. The filter commands are: **type**, "*!*" (not), and **value**.

The values specified in the filter file are compared against the input values. When there is a match, the information is recorded. It is possible to create specific filters for all configured subsystems.

Refer to the man pages for `netfmt` for a complete explanation of the format file syntax.

Filter File Syntax

Each line (filter) in the system level filter file begins with the keyword `FORMATTER FILTER` (for system level filters) or `SX25L3` (for subsystem level filters, level 3 only). Entries are not case-sensitive, and spaces and tabs are ignored. The syntax is shown below:

```
FORMATTER FILTER [type][!] [value]
```

or

SX25L3 `[type][!] [value][value]...`

The *value* field specifies the value for the given *type*. By default, this value is on (applies as a filter), but can be set to off with the "!" (not) prefix. The value used with SX25L3 can be a list of consecutive values, or a range of values.

The permitted *values* for the *types* are given in the table below. See the man pages for `netfmt` for examples.

Table 7-3 Filter File Options

System/subsystem level	Type	Value
FORMATTER FILTER (system level)	kind	`all, hdrin, hdrout, pduin or pduout`
	time_from	12:54:22 (hr:min:sec) 7/25/94 (month/day/year)
	time_through	12:54:22 (hr:min:sec) 7/25/94 (month/day/year)
	subsystem	SX25L3 (level 3)
SX25L3 (subsystem level)	lci	decimal number between 0 and 4095 (this can be a list or range of values)
	packet	`call, callc, clear, clearc, data, rr, rnr, interrupt, reset, resetc, restart, restartc, diagnostic, registration, reject`
	family (refer to Table 7-4 below)	`connect, disconnect, data, fctl, network`

The following table lists the packet types displayed for each *value* combined with *family*.

Table 7-4 Filter File family Options

family	Packet types displayed	
	from DTE to DCE	**from DCE to DTE**
connect	CALL REQUEST CALL ACCEPTED	INCOMING CALL CALL CONNECTED
disconnect	CLEAR REQUEST CLEAR CONFIRMATION	CLEAR INDICATION CLEAR CONFIRMATION
data	DATA DTE INTERRUPT DTE INTERRUPT CONFIRMATION	DATA DCE INTERRUPT DCE INTERRUPT CONFIRMATION
fctl	DTE RR DTE RNR RESET REQUEST DTE RESET CONFIRMATION	DCE RR DCE RNR RESET INDICATION DCE RESET CONFIRMATION
network	RESTART REQUEST DTE RESTART CONFIRMATION	DIAGNOSTIC RESTART INDICATION DCE RESTART INDICATION REGISTRATION

Filter File Examples

The following examples illustrate the use of filter configuration commands within the filter file.

Starts formatting at 10:55:21 on 7/16/94, finishing 3 minutes later:

```
FORMATTER FILTER time_from 10:55:00 7/16/94
FORMATTER FILTER time_through 10:58:00 7/16/94
```

Filters data only on lci numbers 1, 3, 4, 5:

```
SX25L3 lci 1 3-5
SX25L3 packet data
```

After you have created the filter file, check its syntax with:

```
netfmt -p -c filter_file
```

and then run:

```
netfmt -v -c /tmp/filter -f /tmp/tracex25.TRC0 | more
```

strace

The `strace` command collects logging event messages from X.25 level 3 and X.25 level 2 STREAMS modules and writes them to standard output. `strace` runs until terminated by the user. See the man pages for `strace(1)` for a more detailed description.

Running `strace` with several sets of arguments can impair STREAMS performance. Also, some messages may be lost if too many are logged at one time.

HP recommends that you run `strace` as a background process with output directed to a file.

NOTE Only one process can run the STREAMS log driver at a time.

Syntax

```
strace [mod sub pri] ...
```

Parameters

mod Specifies the STREAMS module identifier for:

- X.25 level 3, *mod* should have a value of 200
- X.25 level 2, *mod* should have a value of 201
- X.25 level 1, *mod* should have a value of 210
- IP over X.25, *mod* should have a value of 208
- BSD over X.25, *mod* should have a value of 2503

sub X.25 subnetwork identifier. Use *all* the first time to get subnetwork identifiers (afterwards, you can enter the identifier for a specific subnetwork/interface).

For single-port cards, the code for the *sub* option is the decimal equivalent of *30* + *n* (both expressed in hexadecimal), where *n* is the card instance number. For example, for the interface defined by the device file

x25_1, the *sub* would be 49 (48 + 1 = 49). See the table below for more examples of the codes to be used for specific interfaces.

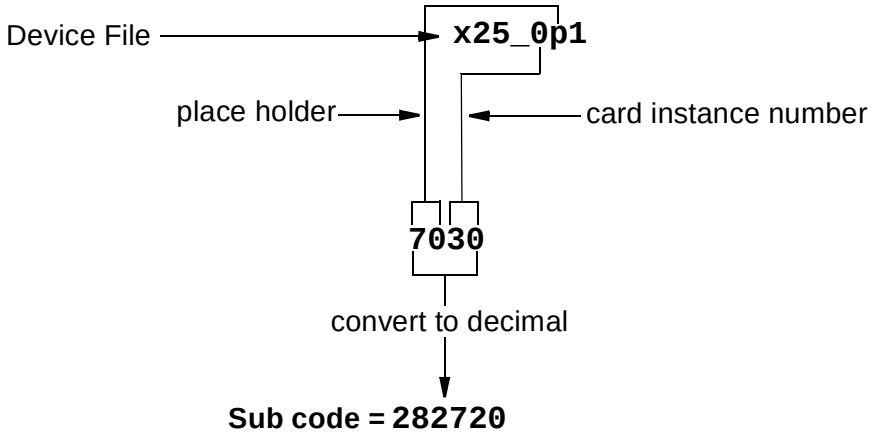
Table 7-5 Single-port Subnet Interface Codes

Card	Calculation (hexadecimal)	Sub code (decimal)
First card = 0 (x25_0)	30 + 0 = 30	48
Second card = 1 (x25_1)	30 + 1 = 31	49
Third card = 2 (x25_2)	30 + 2 = 32	50

For dual-port cards, the *sub* code is the decimal equivalent to the four-digit hexadecimal composite of the place holder code (70) + the card instance number + 30. The example below shows the calculation for the interface defined by the device file x25_0p1.

NOTE The *sub* code for dual-port cards will be the same for both ports.

Figure 7-2



A simple way to think of dual-port *sub* codes is as the decimal equivalent of *7030 + n* (expressed in hexadecimal), where *n* is the card instance number. For example, for the device file x25_1p2, the *sub* in hexadecimal would be

7030 + 1 = 7031

which, when converted to decimal gives you **28721**. Some sample calculations of the *sub* code for dual-port cards are provided in the table below.

NOTE

When specifying the name of device files in systems with dual-port cards, you must use the form

x25_*n*p*x*

where *n* represents the card number (0 to 15, and the first card must be zero), the letter p is a place holder (for port), and *x* represents the interface (port) number (1 or 2).

Table 7-6 Dual-port Subnet Interface Codes

Interface	Calculation (hexadecimal)	Sub code (decimal)
x25_0p1, x25_0p2	7030 + 0 = 7030	28720
x25_1p1, x25_1p2	7030 + 1 = 7031	28721
x25_2p1, x25_2p2	7030 + 2 = 7032	28725
x25_5p1, x25_5p2	7030 + 5 = 7035	28725

pri Specifies the logging priority level. Collects messages from the level equal to or less than the value given by *pri*. Values 1 to 6 are allowed for X.25 level 3, 1 to 4 for level 2, and 1 to 5 for IP over X.25.

The value *all* can be used for any argument in the command line to indicate that there are no restrictions for that argument.

Table 7-7 Output Format

X.25 level 3 (mod=200)	**pri[a]1**	Call/Call Conf (CALL In/Out, CAA)
	pri 2	Clear/Clear Conf (CLR In/Out, CLC)
	pri 3	Reset/Reset Conf (RST In/Out, RSC: Reset Conf)
	pri 4	Restart/Restart Conf (REST In/Out, RESTC)
	pri 5	Interrupt/Int Conf (INT In/Out, INTC)
	pri 6	Data (DATA In/Out)
X.25 level 2 (mod=201)	**pri 1**	Link Up/Down
	pri 2	Link reset events (LINK Rst)
	pri 3	Frame Reject, Reject (FRMR In/Out, REJ In/Out)
	pri 4	Busy conditions (LINK Bsy)
IP over X.25 (mod=208)	**pri 0**	IP over X.25 fatal error (call your HP representative)
	pri 1	IP over X.25 error (check configuration or addresses)
	pri 2	IP over X.25 warning (unexpected event)
	pri 3	IP over X.25 info (IP over X.25 Up/Down)
	pri 4	IP over X.25 trace (internal procedures tracing)
	pri 5	IP over X.25 Data (datagram tracing)

a. pri = Logging priority level

Priorities 4 and 5 for IP over X.25 can only be used if IP over X.25 has been started with the *ifconfig debug* option set

The following table describes the parameters displayed for the various packet types at level 3.

Table 7-8

Packet Type	Parameters Displayed
CALL In/Out	lci number, gfi number
CAA	lci number, gfi number
CLR In/Out	lci number, cause/diagnostic
CLC	lci number
RST In/Out	lci number, cause/diagnostic
RSC:Reset Conf	lci number
REST In/Out	cause/diagnostic
RESTC	no parameter displayed
INT In/Out	lci number
INTC	lci number
DATA	lci number, N(R)/N(S)

The following table describes the parameters displayed for the various frame types at level 2.

Table 7-9

Frame Type	Parameters Displayed
LINK Up	No parameter displayed
LINK Dwn	“Local” means DISC frame sent. “Remote” means DISC frame received.
LINK Rst	“Local” means SABM/UA exchanged, local initiated. “Remote” means SABM/UA exchanged, remote initiated.
FRMR In	No parameter displayed

Table 7-9 (Continued)

FRMR Out	"Rsn 1" means the control field received was not defined or not implemented. "Rsn 3" means the control field received was invalid. "Rsn 4" means the information field received was too long. "Rsn 8" means the control field received contained an invalid variable.
REJ In/Out (level 2)	"N(XX)" is an internal variable of no importance or meaning.
LINK Bsy (level 2)	"Local" means that the RNR has been sent. "Remote" means that the RNR has been received.

Table 7-10 Output Format

X.25 level 1 (mod=210)	**pri[a]0**	Baud control (tx and rx counter)
	pri 0	Baud control (threshold exceeded)
	pri 0	Baud control (number of times threshold exceeded)

a. pri = Logging priority level

Examples of the strace Command

To display all event messages from X.25 level 3 for *all* interfaces, use:

```
strace 200 all all
```

To display all event messages from X.25 level 3 for interface 0, use:

```
strace 200 48 all
```

To display event messages from X.25 level 3 with priority levels 1 to 3, and from X.25 level 2 with priority levels 1 and 2, use:

```
strace 200 all 3 201 all 2
```

Examples of strace Output

The command:

```
strace 200 48 all
```

gives the following output:

```
324112 14:59:16 33683766 1 ... 200 48 CALL Out:‘30’ lci 400 gfi 1

324115 14:59:16 33683776 1 ... 200 48 CAA  In:‘30’ lci 400 gfi 1

324116 14:59:16 33683776 6 ... 200 48 DATA Out:‘30’ lci 400 ns/nr 0000

324119 14:59:16 33683778 6 ... 200 48 DATA In:‘30’ lci 400 ns/nr 0001

324124 14:59:16 33683781 2 ... 200 48 CLR  Out:‘30’ lci 400 C/D 00f2

324127 14:59:16 33683782 2 ... 200 48 CLC  In:‘30’ lci 400
```

Column labels to help you interpret the output are shown in the following table:

Table 7-11

record num	time	tics	pri	ind	mod	sub	data
324112	14:59:16	33683766	1	...	200	48	CALL Out

The command:

```
strace 200 49 all
```

gives the following output:

```
324191 14:59:59 33688076 1 ... 200 49 CALL  In:‘31’ lci 400 gfi 1

324192 14:59:59 33688086 1 ... 200 49 CAA   Out:‘31’ lci 400 gfi 1

324195 14:59:59 33688087 6 ... 200 49 DATA  In:‘31’ lci 400 ns/nr 0000

324196 14:59:59 33688087 6 ... 200 49 DATA Out:‘31’ lci 400 ns/nr 0001
```

```
324203 14:59:59 33688091 2 ... 200 49 CLR   In: ‘31’ lci 400 C/D 00f2

324204 14:59:59 33688091 2 ... 200 49 CLC   Out: ‘31’ lci 400
```

The command:

`strace 201 48 all`

gives the following output:

```
324790 15:06:17 33725841 1 ... 201 48 LINK Up : ‘30’
```

The command:

`strace 201 49 all`

gives the following output:

```
324893 15:07:18 33731984 1 ... 201 49 LINK Dwn: ‘31’ [Remote]
```

NOTE The last example indicates a remote disconnection. ‘30’ and ‘31’ in the examples above are the hexadecimal code for subnetworks 48 and 49 (decimal) respectively.

strerr

Receives error messages from the STREAMS log driver.

The `strerr` daemon receives error messages from the STREAMS log driver (see the man pages for `strlog(7)`). By default, additions to the STREAMS error log files (`error.dd-mm`) are in the STREAMS error log directory (`/usr/adm/streams`).

When first called, `strerr` creates the log file `error.`*`mm-dd`*. This is a daily log file where *`mm`* indicates the month and *`dd`* indicates the day of the logged messages. `strerr` then appends error messages to the log file as they are received from the STREAMS log driver.

NOTE `strerr` should always be used as a background process as it does not return the prompt to the user (the only way to stop it is to kill the process).

Refer to the man pages for `strerr(1)` for a detailed description of the output format.

Syntax

```
strerr [-a sys_admin_mail_name] [-d log_directory]
```

Parameters

`-a sys_admin _mail_name`

Specifies the name of the user who will receive error message by electronic mail.

`-d log_directory`

Specifies the directory that will contain the error log file. The default is `/usr/adm/streams`.

8 Troubleshooting

Troubleshooting Your X.25 Link

This section describes troubleshooting procedures for checking your X.25 link up to X.25 level 3. It also provides information on troubleshooting IP over X.25, as well as procedures for diagnosing switching problems. To use these procedures you should be familiar with the following commands:

Table 8-1 Troubleshooting Commands

Command	Purpose	Refer to...
`x25init`	Initializes X.25 interface and software. Superuser only.	man pages and examples at the end of Appendix B, X.25 Configuration Files and Examples,
`x25stop`	Safely shuts down the interface. Superuser only.	man pages
`x25stat`	Reports on X.25 status. Some options are for superuser only.	man pages and Chapter 5, Diagnostic Utilities, for examples
`x25check` `x25server`	Tests connectivity up to X.25 Level 3 between local and remote nodes.	man pages and Chapter 5, Diagnostic Utilities, for examples
`ping`	Tests IP to IP connectivity.	man pages
`x25ping`	Checks that X.25 interface can reach the remote server.	man pages
`nettl`	Traces and logs X.25 at X.25 packet and LAP B levels	man pages and Chapter 7, Tracing and Logging Utilities, for examples
`netfmt`	Formats trace information collected by `nettl`.	man pages and Chapter 7, Tracing and Logging Utilities, for examples

Table 8-1 Troubleshooting Commands (Continued)

`strace`	Logs X.25 event messages.	man pages and Chapter 7, Tracing and Logging Utilities,
`strerr`	Receives error messages from the Streams log driver.	man pages
`netstat`	Displays configured IP interfaces	man pages
`ifconfig`	Displays IP interface status	man pages

If you still have problems after using the troubleshooting procedures in this section, ask the System Administrator of the remote host to check the X.25 link using the same procedures.

If you have difficulties with high-level network software, refer to the appropriate troubleshooting manual for the particular software product.

This section includes the following troubleshooting flowcharts:

- Hardware check
- X.25 configuration check
- IP over X.25 check

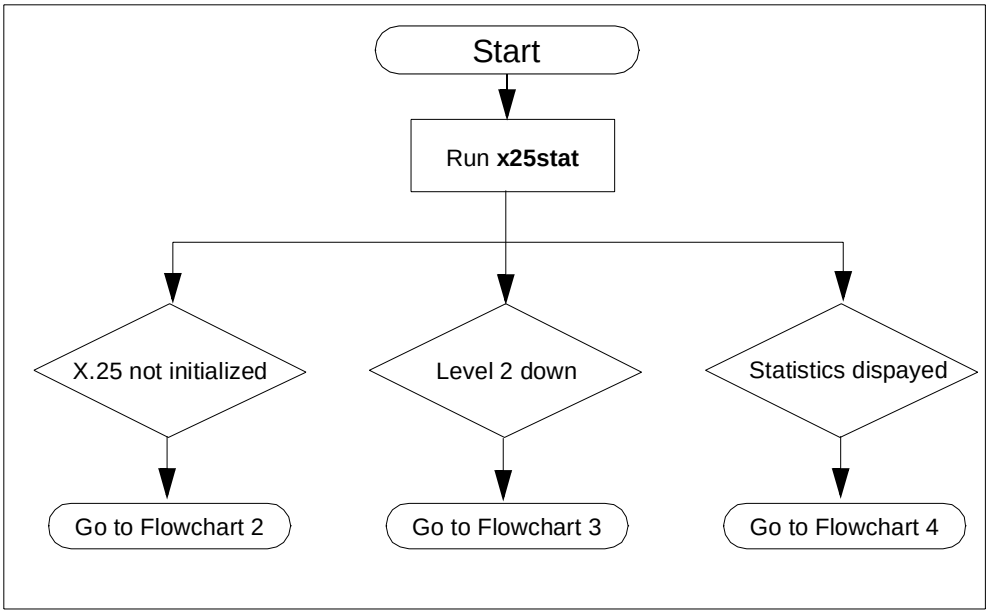
This section also includes information on recovering from a power failure.

Troubleshooting Flowcharts and Procedures

The following pages provide flowcharts and procedures for troubleshooting common problems. Begin with Flowchart 1. If you are running IP over X.25, you should also refer to flowchart 5.

Troubleshooting procedures begin with the `x25stat` command. If **`x25stat -d [devices] -f`** returns a level 2 link state other than “normal,” there is a problem.

Figure 8-1 Flowchart 1



Flowchart 2 – Procedures and Notes

Use these procedures to verify that your X.25 link is correctly initialized. Read the following notes before carrying out the initialization check procedures in Flowchart 2.

Note 2-1 – x25stat

Level 2 is down if **`x25stat -d [device] -f`** returns a level 2 link state other than "normal."

Note 2-2 – eisa_config

Sometimes `x25init` does not recognize the dual-port EISA interface (J2815A) after a first-time installation. When you reboot your system after installing the dual-port EISA interface, you may receive a message similar to:

```
Device /dev/x25_0p1 is not configured in the kernel
```

If this problem occurs, you can work around it by running the `eisa_config` utility in interactive mode as shown below.

Step 1. At the EISA prompt, type:

`add !HWP19A0.CGF <slot number>`

where <*slot number*> is the number of the EISA slot in which the interface card is installed.

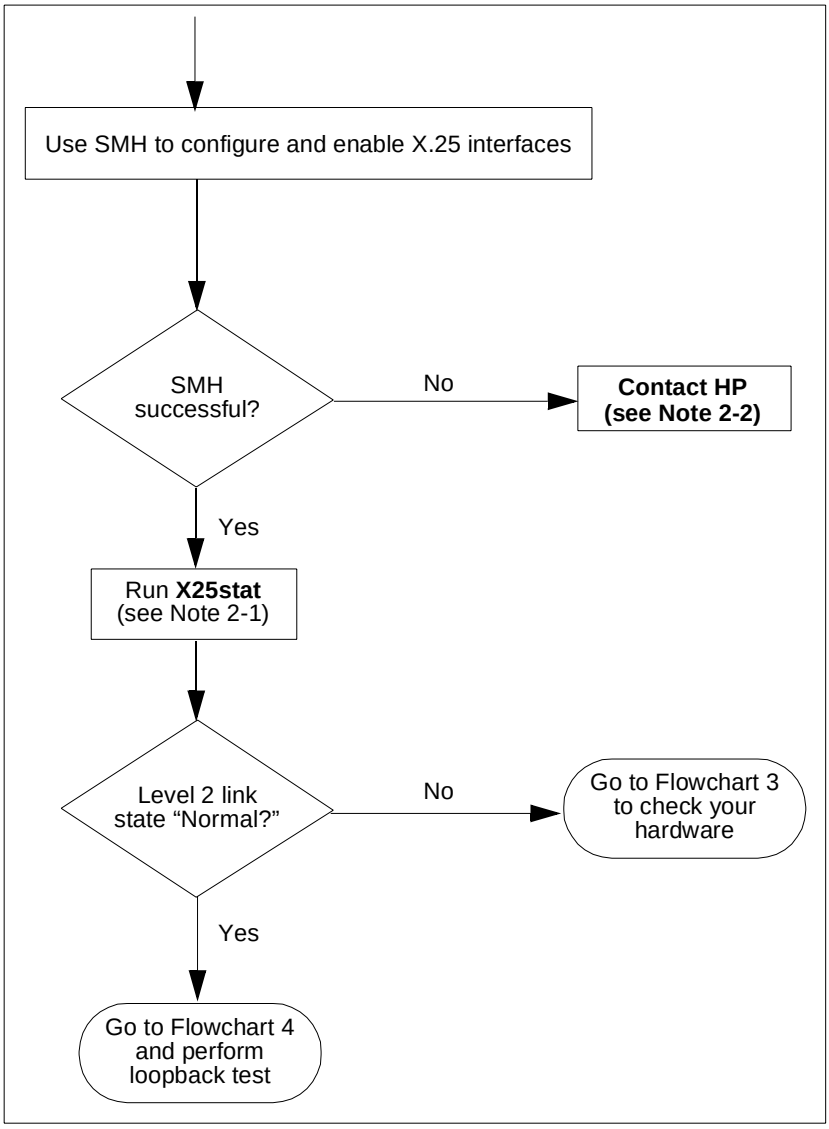
Step 2. Then type:

`save`

`quit`

Step 3. Reboot your system.

Figure 8-2 Flowchart 2 – Initialization Check



Flowchart 3 – Procedures and Notes

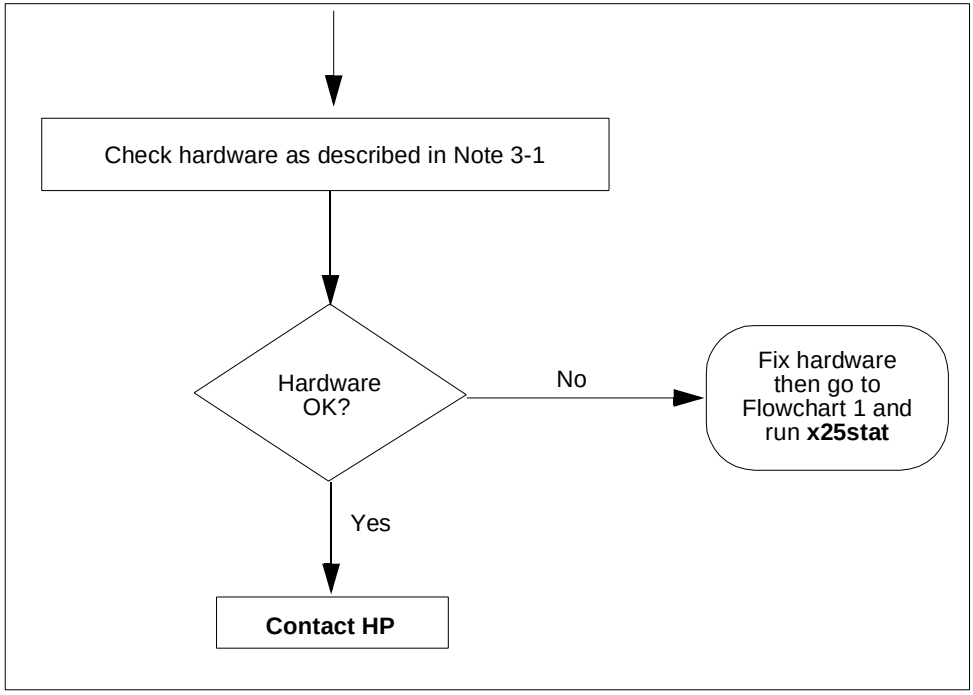
Use these procedures to verify that your hardware is properly connected and operating correctly. Read the following notes **before** running the hardware check procedures in Flowchart 3.

Note 3-1 – Hardware Check

Check the following:

- Interface cable
- Modem
- Network configuration
- Status light (for single port cards only)

Figure 8-3 Flowchart 3 – Hardware Check



Flowchart 4 – Procedures and Notes

This flowchart describes how to run a loopback test using `x25check` on the local node. This checks that your X.25 link to the network or X.25 switch is working correctly.

Note 4-1 – x25check

You do a loopback test using the `x25server` and `x25check` commands. The `x25server` process is only used to accept and echo back an X.25 packet coming from an `x25check` process. To run the loopback test, you need:

- 2 virtual circuits (either 1 one-way inbound and 1 one-way outbound, or 2 two-way),
- if you are connected to a private X.25 switch (rather than connected directly to a network), then the switch must be configured to return a call packet.

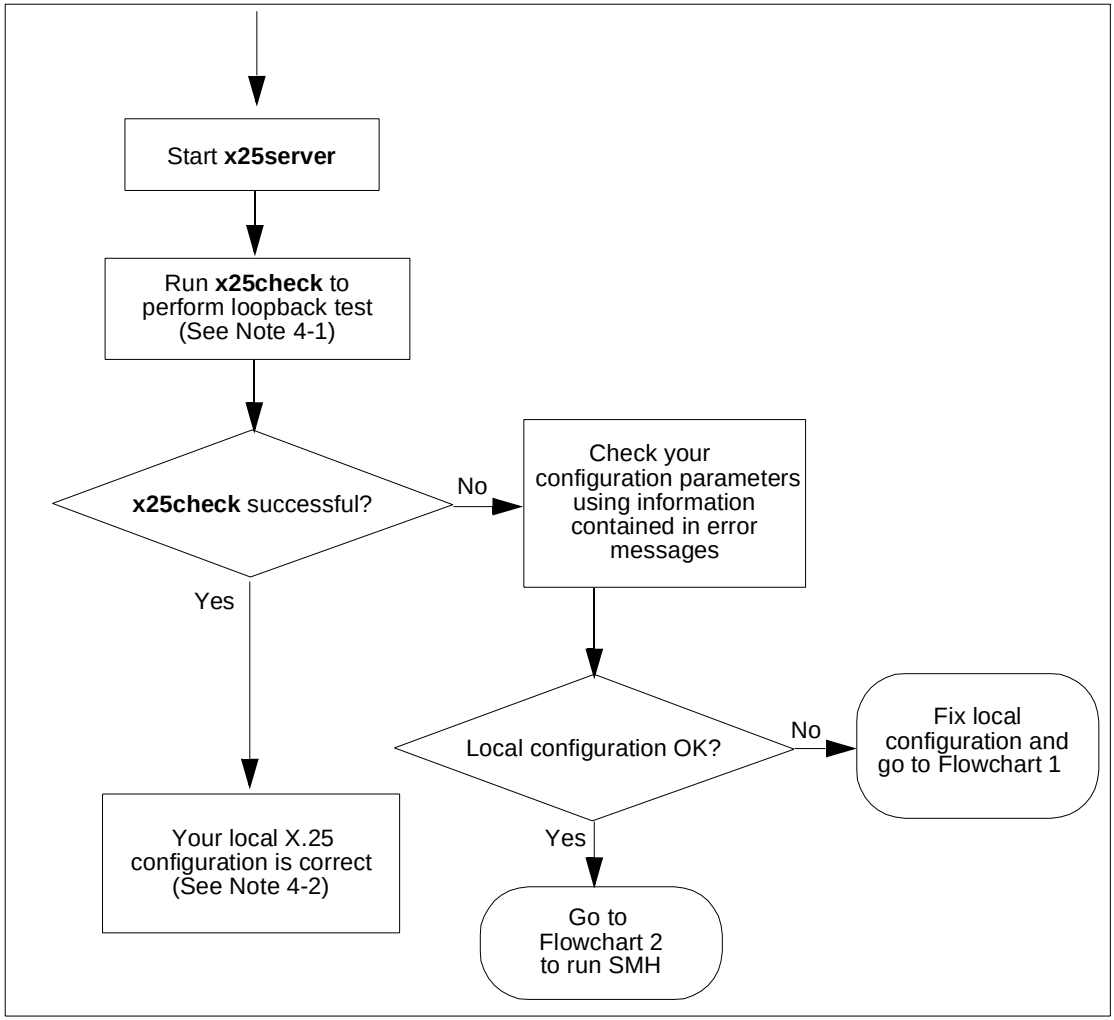
Note 4-2 – Exit

Your local X.25 configuration is correct. You can use `x25check` to verify connections to the remote host only if the remote host is running and:

- has X.25 initialized,
- is running `x25server`,
- has an X.25 configuration that is compatible with the local host's configuration.

If you still experience problems, contact the System Administrator at the remote host and ask them to check their X.25 configuration.

Figure 8-4 Flowchart 4 – X.25 Configuration Check



Flowchart 5 – Procedures and Notes

Use these procedures to verify your IP addressing and remote host connection.

Note 5-1 – ping

Use `ping` on your own IP over X.25 address, for example:

`ping 195.25.0.13`

Do not use aliases (for example, as defined in `/etc/hosts`), since this may introduce other errors.

Use the **Break** key or **Ctrl-C** to terminate `ping` if you have not set the number of packets with the *`-n`* option.

Note 5-2 – Checking your IP over X.25 Configuration

If `ping` returns errors such as "`100% packet loss`," or "`Network Unreachable`," use **`x25stat -a`** to check your address configuration. The most likely causes of problems are:

- No IP address was specified when X.25 was initialized (either in the X.25 configuration file or as a command line option to the `x25init` command).
- IP to X.121 address mapping has not been initialized (using the *`-a`* option with the `x25init` command).
- The IP address is incorrect.
- The mapping of your IP address to your X.121 address is incorrect. Check the IP to X.121 map table file (the default file is `/etc/x25/ip_to_x121_map`).

If no IP communication is possible (`ping` to a local IP address fails) and `netstat -a` shows no entries despite the fact that x25init reported a successful IP mapping, the likely cause is:

- the `device=x25_card_device` parameter has not been added to the `/etc/x25/ip_to_x121_map` file. See the section, "IP-to-X.121 Address Mapping Table" in this chapter.

You can also use the `netstat` command with the *`-r`* option to get information about IP addresses associated with a particular interface.

Note 5-3 – Exit

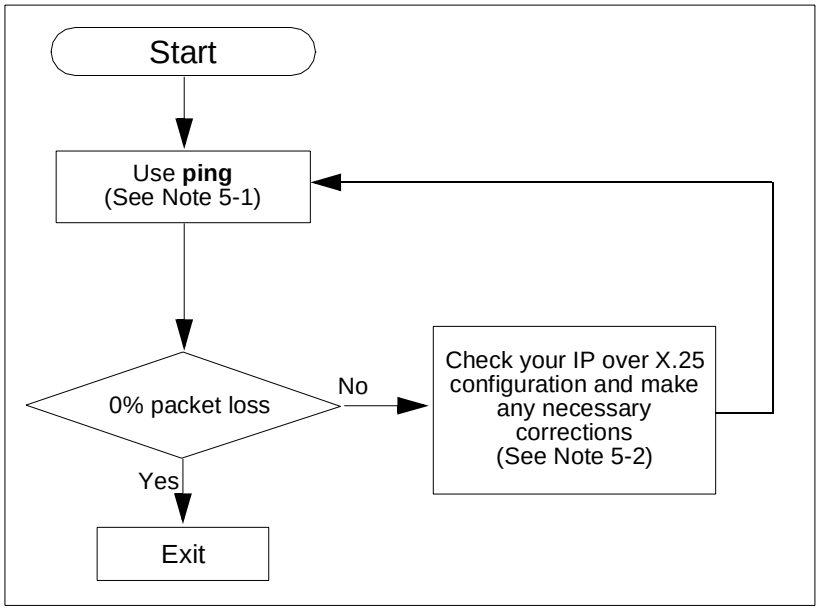
Your local IP over X.25 configuration is correct. You can use `ping` to verify connections if the remote host:

- is operational,

- has X.25 initialized,
- is compatible with IP and ICMP protocols,
- and has an entry for your node in its map table (can map your IP address to your X.121 address).

If you still experience problems, ask the remote host System Administrator to check the IP over X.25 configuration on the remote side of the connection.

Figure 8-5 Flowchart 5 – IP Over X.25 Check



Recovering From a Power Failure

For Systems With a Backup Power Supply

If your site is equipped with a backup power supply, system memory will be preserved in the event of power failure. However, because your X.25 hardware does not have backup capability, any on-board RAM-based memory is lost.

Recovery from a power failure is automatic when the `/etc/powerfail` script is invoked by `/etc/inittab`. Otherwise SMH (or `x25init`) must be used to manually re-initialize the interface after a power failure.

The recovery mechanism informs the X.25 driver of the power failure. The X.25 driver then “marks” the interface as “down” and a DISCONNECT INDICATE is received on all open SVCs.

For Systems With No Backup Power Supply

If your system does not possess the battery backup option, both system memory and interface card memory are lost. You must reboot your system.

Reporting Problems

If you have a service contract with HP, ask your service representative to document the problem as a Service Request (SR).

Include the following information where applicable:

- A detailed description of the problem. Describe the events leading up to the problem and the symptoms of the problem. Include information on HP-UX commands, communication subsystem commands, job streams, result codes, and error messages (the exact wording).
- A record of the output from **`x25 stat -c`** or a copy of the configuration file for each node. Record the output from `netstat`.
- A printout of the existing configuration files.
- A copy of the `ipmap` file.
- A copy of the PSI upload file.
- The version, update, and fix information for all software. From this information HP can determine if the problem is already known and if the correct software is installed at your site.
- Use the `what` command to check your X.25 version.

NOTE Your host node should be running HP-UX Release 10.0 or later. To check the version of your kernel, execute **`uname -r`**.

- Any network log files produced using the `netfmt` command (see Chapter 7, Tracing and Logging Utilities, for details of `netfmt`), and formatted copies of any X.25 trace files that were active when the problem occurred.
- A short description of your application and how it should work.
- **In the event of a system failure, take a full memory dump.** Use the HP-UX utility `/etc/savecore` to save a core dump. Refer *Debugging Streams/UX Modules and Drivers - Streams/UX for the HP 9000 Reference Manual* for details.

- For PAD Problems:
  - — Copies of the `x29hosts` and `x3config` files.
  - — A copy of the output from the **`ls -l /dev/x29`** command if you are running `x29printd` or `x29uucpd`.
  - — Identify which PAD utility is encountering the error (PAD services, PAD emulation, Remote PAD printer, or UUCP). Include all files relative to that service (for example, `.login`, `profile ID`, `/usr/spool/lp/*`, or `/usr/lib/uucp/*`).
  - — Include log files for the PAD services.
  - — Provide a detailed description of the problem.

For more information about files related to PAD services see Chapter 6, “PAD Services.”

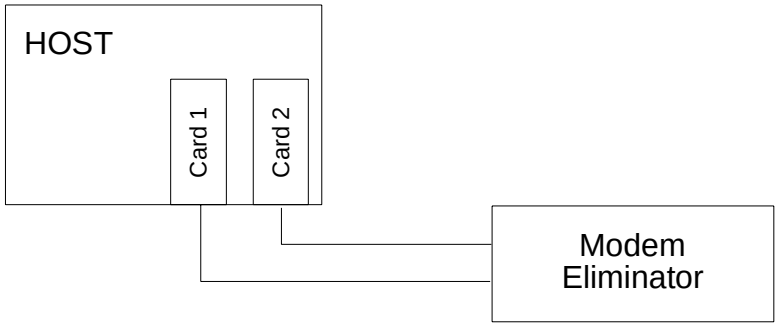
NOTE If you do not have a service contract with HP, you can still follow the procedure described above, but you will be billed accordingly for time and materials.

Back-to-back Configuration on the Same Host

Systems with multiple X.25 interfaces can use back-to-back configurations to perform diagnostics. If you suspect that a switch might be the source of your connection problem, you can set up a back-to-back configuration to test the circuit without the PAD switch.

To set up a back-to-back connection on a single host, you connect one card to another card on the same host through a modem eliminator, as shown below. The modem eliminator provides the necessary clocking for the DCE. The most common modem eliminators are RS-232 and V.35.

Figure 8-6



In order for two cards (on the same host) to communicate, one card must be configured as the DTE and the other card as DCE.

To set up a card as DCE, use the `x25init` command and enter *`DCE_80`*, *`DCE_84`* or *`DCE_88`* (according to the standard you are using) in the *`networktype`* parameter.

You must also make sure that the two cards are using compatible level 2 parameters.

Once both cards are initialized, you can run `x25check` and `x25server` to verify the connections. If `x25check` completes successfully in this setup but does not complete successfully with the switch in place, the switch is incorrectly configured. Check with your network provider for correct switch configuration.

Configuration and Troubleshooting Commands

This section provides a brief summary of X.25 commands and indicates where further information can be found. Examples of `x25init` and the syntax for IP over X.25 mapping are provided at the end of the section.

Command Summary

NOTE You can use the `x25message` utility to get an explanation of an error message. Enter the command at the HP-UX prompt followed by the text of the message, and the system will display an explanation of the cause along with recommendations for corrective action.

`x25init`

Initializes your X.25 link (requires Superuser privileges). See your man pages and the examples later in this section.

`x25stop`

Gracefully shuts down an X.25 link (requires Superuser privileges). See your man pages.

`x25check and x25server`

Together, these commands can be used to test connectivity between two nodes up to the X.25 access level (level 3). See your man pages and the examples in Chapter 5, “Diagnostic Utilities.”

`x25stat`

Displays X.25 link status, configuration information, and virtual circuit statistics. See your man pages and the examples in Chapter 5, “Diagnostic Utilities.” Some options require Superuser privileges.

`x25mibstat`

Utility with programmatic interface that allows users to get additional X.25 statistics.

x25ping

Checks if a remote host can be reached via the X.25 interface. See your man pages.

ping

Tests connectivity to a remote host up to the Internet Protocol (IP) level (level 3). See your man pages.

route

Used to add a host or network to the network routing table. See your man pages.

proxy

The Probe proxy table is the NS equivalent of the Internet/Berkeley Services `/etc/hosts` file. See your man pages.

netstat

Displays network statistics and information about network connections. See your man pages.

x25trace

Protocol level tracing facility. See your man pages.

x25upload

Dumps the interface card memory into a file. See your man pages.

nettl

Creates a user-defined log file containing frame and packet trace information from X.25 packet and LAP-B levels. See your man pages for `x25trace` and the examples in Chapter 7, “Tracing and Logging Utilities.”

netfmt

Formats trace information contained in the user-defined log file that was generated with the `nettl` command. See your man pages for `x25trace` and the examples in Chapter 7, “Tracing and Logging Utilities.”

strace

Collects and writes X.25 logging event message. See your man pages and the examples in Chapter 7, "Tracing and Logging Utilities."

strerr

Receives error messages from the STREAMS log driver. See your man pages and the examples in Chapter 7, "Tracing and Logging Utilities."

ifconfig

Configures network interface parameters. See your man pages.

Examples of x25init

The following example shows a valid `usr/sbin/x25init` command (provided that the programmatic access name has been specified in the configuration file `x25config_0`):

```
x25init -c x25config_0 4085551111 -a
ip_to_x121_map
```

Alternatively, the IP-to-X.121 map table can be specified in a separate `x25init` command:

```
x25init -c x25config_0 4085551111
x25init -a ip_to_x121_map
```

NOTE If `x25init` fails, you can look at the `/usr/adm/x25/x25init.log` log file for troubleshooting information.

IP-to-X.121 Address Mapping Table

The IP-to-X.121 address mapping table is used to enable IP to route through X.25 networks by mapping a host's IP address to its X.121 address. The table is used for both inbound and outbound packets.

The IP-to-X.121 address mapping table is a kernel table created from an ASCII file containing one entry per line. It is created using `x25init` with the *-a* option. Each line describes the destination X.121 address used to

reach a host with a particular IP address. Reverse mapping reverses the order of translation. The default file name and directory for the mapping table is `/etc/x25/ip_to_x121_map`.

When the packet is outbound, the IP address is mapped to the X.121 address that is associated with it in the IP map file. When the packet is inbound, the X.121 address is mapped to the IP address of the same destination host. This is called "reverse mapping." The table is not used when IP packets are transmitted to X.25 over DDN. In this case, the DDN-specific mapping algorithm is used instead. See *DDN X.25 Host Interface Specification (BBN83)*.

There must be an entry in the mapping table for each host you want to communicate with over a direct-connect X.25 network. If the host is connected through a gateway, you only need an entry for the gateway (containing the IP and X.121 addresses of the gateway).

NOTE If the destination host is only reachable through a gateway, you must have a route to that host (a host or a net route).

The IP address in the table must belong to the same IP subnet as a configured card in your system, and this card must be already started when you initialize IP routing with the command:

```
/usr/sbin/x25init -a /etc/x25/ip_to_x121_map
```

If the IP address does not correspond to a configured card, the X.25 product does not know which interface to use to send packets. Therefore, you must place this information in your `/etc/x25/ip_to_x121_map` file by adding the *x25_card_device* parameter as shown in the syntax example below. **This is a manual procedure that cannot be done in SMH**.

Syntax for SVCs

```
IP_address  X.121_address [{+URC|-URC}][+ARC|-ARC}] device=x25_card_device,
packet_size, window_size [CUG=n]
```

Syntax for PVCs

```
IP_address  X.121_address pvc=n name=prog_access_name
```

NOTE If you do not include the *packet_size* and *window_size* values, and the *x25_card_device* is the last field, the default values will be applied to the *packet_size* and *window_size* fields.

Example

```
15.128.174.04 12345 -URC -ARC

30.0.0.2 34567 -URC -ARC device=x25_0,256,7
```

In this example, a card has been configured with a device equal to x25_0, a packet size of 256, and a window size of 7.

You can then use the `gated` or `route add` command to send datagrams to the IP address, 30.0.0.2, routed through your X.25 network to the address, 34567. These packets will be sent out through the X.25 card that corresponds to the deveice `x.25_0`.

File Mapping Parameters

IP_address Specifies a valid IP address in standard dot notation form:

n.n.n.n

where *n* is a number from 0 to 255 inclusive.

X.121_address Specifies the corresponding X.121 address for the host with the above IP address.

pvc=n Specifies which permanent virtual circuit should be used for the X.121 destination. The value *pvc=* is a keyword set to *n*, which is an integer variable indicating the lci of the permanent virtual circuit. This keyword/variable pair is specified only when a permanent virtual circuit is to be used for IP access. When it is not specified, the default value of a two-way SVC is assumed.

The following information is provided for reference only.

Because *pvc* describes a permanent circuit for a particular interface, that interface must have been initialized prior to the initialization of the address map table. If the interface or the specified PVC does not exist, an error is reported.

prog_access_name Specifies the card's programmatic access name.

NOTE For Series 712 workstations this is always *name=interface0*.

+URC Specifies that outbound calls to this IP address will Use Reverse Charging (URC).

-URC (Default) Specifies that outbound calls to this IP address will not Use Reverse Charging (URC).)

+ARC Specifies that inbound calls to this IP address will Accept Reverse Charging (ARC).

-ARC (Default) Specifies that inbound calls to this IP address will not Accept Reverse Charging (ARC).

x25_card_device A device configured for one of your existing X.25 cards. This information is used by the IP_to_X25 software to identify the interface port through which the outbound packets will be sent.

CUG=n Specifies the Closed User Group index number used for placing calls to this IP address. The index number is a two- or four-digit number. Two digits corresponds to the basic format, and four digits corresponds to the extended format of the CUG selection facility.

You can put comments in your IP mapping table by including the # character at the beginning of the line. Note that you cannot add comments after a table entry.

NOTE For non-DDN configured interfaces, all remote hosts must have entries in the IP-to-X.121 address map table. Remote hosts not on your X.25 network (that is, on the other side of gateways) do not need to have map table entries. For more information, refer to the routing description in this section.

A Using Non-English Subscription Forms

Subscription Form Translations

This appendix lists the English (SMH dialog field) equivalents for terms that appear on your French, Italian, German or Spanish subscription form. Use this information to locate the English SMH dialog field equivalents of these terms. Terms are listed in order of appearance. Online help (using the **Help** button) is available for each field.

Table A-1 French TRANSPAC Subscription Form

English on SMH dialogs	French on subscription form
X.25 Address	No Transpac
Network Carrier Type	TRANSPAC
Permanent - *Quantity*	Nombre V.L. CVP
Switched (inbound) - *Quantity*	Nombre V.L. CVC Arrivée
Switched (two-way) - *Quantity*	Nombre V.L. CVC Mixte
Switched (outbound) - *Quantity*	Nombre V.L. CVC Départ
Fast Accept Selected	Acceptation sélection rapide
Flow Control Negotiation	Nég. taille Paquet et Fenêtre
Reverse Charge Accepted	Acceptation, PVC (TAD)
Default Packet Size - *Inbound & Outbound, Switched VC Flow Control*	Longueur paquet
Default Window Size - *Inbound & Outbound, Switched VC Flow Control*	Taille de la fenêtre
Packet Size - *Permanent VC Flow Control*	Longueur paquet
Switched VC Default - *Inbound & Outbound*	Choix classe de débit
Switched VC Negotiated	Négociation classes de débit

Table A-1 French TRANSPAC Subscription Form (Continued)

k - Level 2 Window Size (frames)	Fenêtre
T1 - Retransmission Timer (ms)	Temporisateur

Table A-2 Using Your ITAPAC (Italian) Subscription Form

English on SMH dialogs	Italian on subscription form
X.25 Address	Indirizzo X.25
Network Carrier Type	ITAPAC
Permanent - *Quantity*	Circuiti permanenti-numero
Switched (inbound) - *Quantity*	Circuiti commutati unidirectionali in entrata - numero
Switched (two-way) - *Quantity*	Circuiti commutati bidiretionali in entrata - numero
Switched (outbound) - *Quantity*	Circuiti commutati unidirectionali in uscita- numero
Fast Accept Selected	Selectione rapida
Flow Control Negotiation	Negoziazione dei parametri di controllo di flusso
Reverse Charge Accepted	Accettazione delle tassa zione al chiamato
Default Packet Size - *Inbound & Outbound, Switched VC Flow Control*	Dimensione di default del pacchetto - uscente, controllo di flusso VC commutato
Default Window Size - *Inbound & Outbound, Switched VC Flow Control*	Dimensione di default della finestra - uscente, controllo di flusso VC commutato
Packet Size - *Permanent VC Flow Control*	Dimensione del pacchetto - controllo di flusso VC permanente

Table A-2 Using Your ITAPAC (Italian) Subscription Form (Continued)

Switched VC Default - *Inbound & Outbound (Modify Throughput Class Setting)*	Classe di throughput di default in entrata & in uscita
Switched VC Negotiated (*Modify Throughput Class Setting*)	Classe di throughput negocia
k - Level 2 Window Size (frames)	k - Dimensione di finestra a livello 2
T1 - Retransmission Timer (ms)	T1 - Timer di ritramissione (ms).

Table A-3 Using your DATEX-P (German) Subscription Form

English on SMH dialogs	German on subscription form
X.25 Address	Rufnummer des Wählanschlusses
Network Carrier Type	DATEX-P
Permanent - *Quantity*	Feste virtuelle Verbindungen
Switched (inbound) - *Quantity*	Nur ankommende
Switched (outbound) - *Quantity*	Nur abgehend
Switched (two-way) - *Quantity*	Abgehend und ankommend
Reverse Charge Accepted	Möglichkeit der Verbindungsgebühreüber nahme bei ankommenden gewählten Ruf
Default Window Size - *Inbound & Outbound, Switched VC FlowControl*	Fenstergrösse w Senden Fenstergrösse w Empfangen

Table A-4 Using your IBERPAC (Spanish) Subscription Form

English on SMH dialogs	Spanish on subscription form
X.25 Address	NRI (Número Red IBERPAC)
Network Carrier Type	Tipo de Red (IBERPAC)

Table A-4 Using your IBERPAC (Spanish) Subscription Form (Continued)

Permanent - *Quantity*	Canales Lógicos Permanentes - cantidad
Switched (inbound) - *Quantity*	Canales Lógicos Unidireccionales Entrante - cantidads
Switched (two-way) - *Quantity*	Canales Lógicos Bidireccionales - cantidad
Switched (outbound) - *Quantity*	Canales Lógicos Unidireccionales Salientes - cantidad
Fast Accept Selected	Selección Rápida
Flow Control Negotiation	Negociación del Control de Flujo
Reverse Charge Accepted	Cobro Revertido
Default Packet Size - *Inbound & Outbound, Switched VC Flow Control*	Tamaño de Paquete por defecto - entrante y saliente, control de flujo en CV conmutados
Default Window Size - *Inbound & Outbound, Switched VC Flow Control*	Tamaño de Ventana por defecto - entrante y saliente, control de flujo en CV conmutados
Packet Size - *Permanent VC Flow Control*	Tamaño de Paquete - control de flujo en CV permanentes
Switched VC Default - *Inbound & Outbound (Modify Throughput Class Setting)*	CV conmutados por defecto - entrantes y salientes (Negociación de la Clase de Caudal)
Switched VC Negotiated (*Modify Throughput Class Setting*)	Clase negociada en CV conmutado (*Negociación de la Clase de Caudal*)
k - Level 2 Window Size (frames)	k - Tamaño de Ventana en Nivel 2 (tramas)
T1 - Retransmission Timer (ms)	T1 - Temporizador de Retransmisión (ms)

B X.25 Configuration Files and Examples

X.25 Configuration Files

This section lists X.25 configuration and addressing files.

Table B-1 /etc/x25 directory

`x25config_0`	ASCII file containing X.25 parameters. This file can be used by the `x25init` command when initializing the card. It is created automatically if you configure X.25 using SMH. To configure X.25 manually, copy and edit either `x25init_def` or `x25init_smpl`.
`x25init_def`	Contains default X.25 configuration parameters and example configuration parameters for parameters that have no default. The listing for this file is shown later in this chapter. This file is read-only.
`x25init_smpl`	Contains example X.25 configuration parameters. The listing for this file is shown later in this chapter. This file is read-only.
`ip_to_x121_map`	Mapping table between X.25 and IP addresses. See “Configuration and Troubleshooting Commands” in Chapter 8, Troubleshooting, for more information on this file.
`x25_networks`	Contains information describing several canonical network types. The listing for this file is shown later in this chapter. This file is read-only.

Table B-2 /etc directory

`hosts`	Associates Internet addresses with official host names and aliases. See the man page for `hosts`.
`hosts.equiv (optional)`	Security file which authorizes remote hosts and users on local host. See the man page for `hosts.equiv`.

Table B-2 /etc directory (Continued)

`networks`	Contains information regarding known networks.

Table B-3 /var/x25/log/x25server directory

`x25server.log`	Log file of output from `x25server` process.

Table B-4 `/var/x25/log` directory

`x25init.log`	Log file of output from `x25init` command.

Table B-5 `Home` directory

`.rhosts` (optional)	Security file which authorizes remote hosts and users on local host. See the man page for `hosts.equiv`.
`.netrc` (optional)	Contains login and initialization information used by the ftp auto-login process. See the man page for `netrc`.

Example Files

This section provides example file listings for of the `x25init_def`, `x_25init_smpl`, `x3config`, `x29hosts`, and `x25_networks` files.

The x25init_def File

Table B-6

```
#
# Likely runstring: x25init -c x25init_def
# This sample X.25 configuration contains default values

# Mandatory parameter - X.121 address
# X.121 4085551202

# Typically the address on the packet is identical to X.121
# packet address except for some networks like TransPac which
# require that we put null address on the packet.

# X.121_packet ''      # use a null (i.e. length 0) packet

# Mandatory parameter - interface name for Level 3 access
# name interface0

# Mandatory parameter - device to initialize
# device x25_0p1

# Level 2 Parameters

t1 3000               # frame retransmission timeout; 3 seconds

t3 60000              # Level 2 idle timer; 60 seconds

# The frame size varies depending on flow-control negotiation
# and whether you subscribe for fast-select feature or not.
# max Level 2 transmission size (octets) [n1/8] for packet size
# 128 with fast select disabled is 149.

framesize 149         # n1/8 octets
n2 20                 # number of retransmissions allowed
k 7                   # Level 2 window
```

Table B-6 (Continued)

```
# Level 3 Parameters

# Mandatory Parameters - virtual circuit parameters
# logical channel id, start num [1-4095], type, how many
# At least one lci type (PVC, incoming, two-way or outgoing SVC)
# needs to be configured.

#lci 1 pvc 5            # 5 permanent VCs starting at LCI 1
#lci 255 insvc 5        # 5 one-way incoming SVCs starting at LCI
#lci 2048 svc 6         255
#lci 3072 outsvc 6      # 6 two-way switched SVCs starting at LCI
                        2048
                        # 6 one-way outgoing SVCs starting at LCI
                        3072

networktype DTE_84      # CCITT 1984, DTE (see
fast_select disabled    /etc/x25/x25_networks)
reverse_charge          # disallow incoming calls with call user
disabled                data
                        # disallow incoming calls requesting
                        reverse changes

def_inpacketsize 128    # default packetsize
def_outpacketsize 128   # default packetsize
def_inwindow 2          # default window size
def_outwindow 2         # default window size
# def_inthruputclass    # default thruput class
19200                   # default thruput class
# def_outthruputclass
19200

flowcontrol off         # flow control negotiation not allowed
neg_inpacketsize 128    # offered pkt size if using flow control
neg_outpacketsize 128   negotiation
neg_inwindow 2          # offered pkt size if using flow control
neg_outwindow 2         negotiation
                        # offered wndw size if using flow control
                        negotiation
                        # offered wndw size if using flow control
                        negotiation

thruputclass off
```

Table B-6 (Continued)

```
# neg_inthruputclass    # offered thruput class if using thruput
19200                   # class negotiation
                        # offered thruput class if using thruput
# neg_outthruputclass   # class negotiation
19200

pvc_inpacketsize 128    # packetsize for PVCs
pvc_outpacketsize 128   # packetsize for PVCs
pvc_inwindow 2          # window size for PVCs
pvc_outwindow 2         # window size for PVCs

# These set of parameters are specified only if you wish to run
IP over X.25.
# IP Related Parameters

# IP 15.4.64.120
255.255.248.0           # IP address and subnet mask
# mtu 2048              # max transmission unit 2048 octets for
# mtu 1007              standard
hold 300                # max transmission unit 1007 octets for
idle 600                DDN
                        # 5 minute hold timer
                        # 10 minute idle timer#
```

The x25init_smpl File

Table B-7

```
#
# Likely runstring: x25init -c x25init_smpl

X.121 4085551202        # X.121 address
X.121_packet ''         # use a null (i.e. length 0) packet
                        # address (TransPac addressing)
name interface0         # interface name for Level 3 access
device x25_0p1          # device to initialize

# Level 2 Parameters

t1 3000                 # frame retransmission timeout; 3
t3 12000                seconds
framesize 263           # Level 2 idle timer; 12 seconds
n2 3                    # max Level 2 transmission size
k 7                     (octets) [n1/8]
                        # number of retransmissions allowed
                        # Level 2 window

# Level 3 Parameters

# virtual circuit parameters
# logical channel id, start num [1-4095], type, how many

lci 1 pvc 5             # 5 permanent VCs
lci 255 insvc 5         # 5 one-way incoming SVCs
lci 2048 svc 6          # 6 two-way switched VCs

networktype TRANSPAC    # CCITT 1984, DTE (see
fast_select enabled     /etc/x25/x25_networks)
reverse_charge          # allow incoming calls with call user
enabled                 data
                        # allow incoming calls requesting
                        reverse changes

def_inpacketsize 128    # default packetsize
def_outpacketsize 128   # default packetsize
def_inwindow 7          # default window size
def_outwindow 7         # default window size
def_inthruputclass      # default thruput class
19200                   # default thruput class
def_outthruputclass
19200
```

Table B-7 (Continued)

```
flowcontrol on          # flow control negotiation allowed
neg_inpacketsize 128    # offered pckt size if using flow
neg_outpacketsize 128   control negotiation
neg_inwindow 7          # offered pckt size if using flow
neg_outwindow 7         control negotiation
                        # offered wndw size if using flow
                        control negotiation
                        # offered wndw size if using flow
                        control negotiation

thruputclass on
neg_inthruputclass      # offered thruput class if using
19200                   thruput class negotion
                        # offered thruput class if using
neg_outthruputclass     thruput class negotiation
19200

pvc_inpacketsize 128    # packetsize for PVCs
pvc_outpacketsize 128   # packetsize for PVCs
pvc_inwindow 7          # window size for PVCs
pvc_outwindow 7         # window size for PVCs

# IP Related
Parameters

IP 15.4.64.120
255.255.248.0           # IP address and subnet mask
mtu 2048                # max transmission unit 2048 octets
hold 300                # 5 minute hold timer
idle 600                # 10 minute idle timer
```

The x3config File

```
hp_printer {
      1      0
      2      0
      3      0
      4      10
      5      1
      6      0
      7      0
      8      0
      9      0
      10     0
      11     14
      12     1
      13     0
      14     0
      15     0
      16     8
      17     24
      18     0
      19     1
      20     0
      21     0
      22     0
}
hp_uucp {
      1      0
      2      0
      3      0
      4      10
      5      1
      6      0
      7      0
      8      0
      9      0
      10     0
      11     14
      12     1
      13     0
      14     0
      15     0
      16     8
      17     24
      18     0
      19     1
```

```
      20     0
      21     0
      22     0
}
hp_padsrvr {
      1      1      1
      2      1      1
      3      94     127
      4      0      0
      5      1      1
      6      5      5
      7      21     21
      8      0      0
      9      0      0
      10     0      0
      11     14     14
      12     1      1
      13     0      0
      14     0      0
      15     1      0
      16     8      8
      17     24     24
      18     0      0
      19     1      1
      20     0      0
      21     0      0
      22     0      0
}
hp_profile : 0 {
      1      1
      2      1
      3      127
      4      0
      5      1
      6      5
      7      21
      8      0
      9      0
      10     0
      11     14
      12     1
      13     0
      14     0
      15     1
      16     8
      17     24
```

```
        18      0
        19      1
        20      0
        21      0
        22      0
}
```

The x29hosts File

Table B-8

```
# for x29printd
printer {

          device           printer1
          name             hptndxk0
          remote_x121      408555111201
          x3               hp_printer
          reverse_charge   enable
          logging          1

}

# for x29uucpd
pad_uucp {

          device           x25uucp
          name             hptndxk0
          remote_x121      4085551113
          x3               hp_uucp
          reverse_charge   enable
          logging          3

}

# for padem
pad_em {

          name             hptndxk0
          remote_x121      4085551111
          reverse_charge   enable
          profile          0
          logging          3

}

host_table {

          Gale             4085551111
          Tornado          4085551113
          Typhoon          4085551115
```

Table B-8 (Continued)

```
}

# for x29server
pad_spt {

          remote_x121      408555120801
          x3               hp_padsrvr
          logging          1
          reverse_charge   disable

}
```

The Network Type File (x25_networks)

The network type file describes standard network types. If the network to which you are subscribing does not meet any of the network descriptions specified in this file, use one of the generic entries (DTE_80, DTE_84 or DTE_88). The network type file is read-only.

The value in the first column of the file is an alias assigned to the network type. This is the value that you specify for the *networktype* configuration parameter in the `x25init` configuration file. The value in the second column is the actual network type. The value in the third column is the version of the CCITT X.25 Recommendation with which the network complies: 1980, 1984, or 1988.

The delivered X.25 `networktype` file, `/etc/x25/x25_networks` is shown below:

Table B-9

```
DTE_80              L3_DTE              1980
DTE_84              L3_DTE              1984
DTE_88              L3_DTE              1988
DCE_80              L3_DCE              1980
DCE_84              L3_DCE              1984
DCE_88              L3_DTE              1988
AUSPAC              AUSPAC              1984
DATANET1            DATANET1            1984
DATAPAC             DATAPAC             1980
DATEXP_AUSTRIA      DATEXP_AUSTRIA      1984
DATEXP_DEUTSCHE     DATEXP_GERMANY      1980
DCS                 DCS                 1980
DDN                 DDN_NET             1980
DDXP                DDXP                1980
HPPPN               HPPPN               1984
ITAPAC              ITAPAC              1980
LUXPAC              LUXPAC              1980
PSS                 PSS                 1980
TELENET             TELENET             1980
TRANSPAC            TRANSPAC            1984
TYMNET              TYMNET              1980
```

An alias associated with a network type chosen from this file is used to identify the network type during the configuration process.

C Diagnostic Messages

Introduction

This appendix describes the diagnostic codes and messages which are a subset of the list defined by the International Standards Organization (ISO) in *IS-8202:1987(E)*. Only those diagnostics supported on X.25 for HP 9000 systems are listed. Each message is listed in numerical order by its diagnostic code and includes an ISO description and explanation.

Refer to Appendix D of the *X.25 PSN Connection* for a complete list of diagnostic codes and messages, and to *ISO IS-8202:1987(E)* for a list of the corresponding cause codes and messages.

Diagnostic Message Example

An example diagnostic message generated by the network from a CLEAR REQUEST packet is shown below:

```
Clear request
  LCGN: 0               LCN : 32
  Clearing Cause [0] : DTE originated.
  Diagnostic [241] :
```

In the example above, LCGN is the Logical Channel Group Number and LCN is the Logical Channel Number (the number denoting the logical association between a DTE and DTE connected by a VC). Refer to the discussion below for more information on cause codes.

Diagnostic codes are generated by the network and may appear in octet 5 of RESTART INDICATION, RESET INDICATION, or CLEAR INDICATION packets. If a diagnostic field is not present in a CLEAR INDICATION or RESET INDICATION packet, 0 is returned as the diagnostic code.

A cause code and message is usually returned from the network with each diagnostic. Diagnostic codes and messages are usually displayed only when tracing or logging is on, the `x25check` utility is run, or cause and diagnostic codes are printed in level 3 application programs. Refer to the listing of diagnostic codes later in this appendix for more information.

In the descriptions in the diagnostic listing, a REQUEST or ACCEPTED packet refers to a packet generated by the local application and an INDICATION or CONFIRM packet refers to a packet received from the network. The requests in which a diagnostic code can be found are shown after the numeric code of the diagnostic.

Once the application specifies CLEAR or RESET cause and diagnostic codes, this code combination is used for all RESET REQUEST packets generated by the application. This combination is also used for the CLEAR REQUEST generated when the application calls `close()` or `shutdown()` on a given socket.

Cause Code Settings

X.25 automatically resets (silently forces) the network-generated cause code of CLEAR or RESET REQUEST packets to comply with the applicable *CCITT X.25 Recommendation*.

- 1980 – The cause is silently forced to zero.
- 1984 – The cause is left at zero or OR'ed with 128 if it is not zero.
- 1988 – The cause is left at zero or OR'ed with 128 if it is not zero.

X.25 does not guarantee that the cause, diagnostic, facility, or clear user data fields of a CLEAR REQUEST or a RESET REQUEST are delivered to the remote destination because these fields may be altered by the network, the X.25 implementation (under certain circumstances) or both. In particular, X.25 discards the clear user data and facility fields of a CLEAR REQUEST when the facility field is found to be invalid for the given system configuration or when any other error occurs. When an error occurs on a CLEAR REQUEST, the cause and diagnostic fields delivered to the network are silently set according to the error.

Packet Codes

RESTART Packet Codes

When the system receives a RESTART INDICATION packet, the received CLEAR INDICATION or RESET INDICATION packet contains a cause code of 0 and the diagnostic code used in the RESTART.

RESET/CLEAR Packet Codes

The system may generate a RESET/CLEAR because of an error, lack of memory, or other condition. When this occurs, a RESET/CLEAR INDICATION is delivered to the application and a RESET/CLEAR REQUEST packet is sent to the network. The cause code is always zero as dictated by ISO in the IS-8208 standard, and the diagnostic is one of the codes in this appendix. Both the REQUEST and the INDICATION carry the same cause/diagnostic combination.

X.25 Diagnostic Messages

Table C-1 No Additional Information

Diagnostic	ISO Description	Explanation
1 (RESET)	Invalid P(S).	DATA packet received from network with P(S) invalid or outside of allowable window.
2 (RESET)	Invalid P(R).	DATA packet received from network with P(R) invalid.

Table C-2 Packet Type Invalid

Diagnostic	ISO Description	Explanation
17 (RESTART, CLEAR, RESET)	Packet type invalid for state R1.	Invalid packet received with interface in state R1 (packet level ready). RESTART REQUEST is sent on network. Invalid packet is probably RESTART CONFIRMATION.
20 (CLEAR)	Packet type invalid for state P1.	Invalid packet received on VC in state P1 (Ready).
21 (CLEAR)	Packet type invalid for state P2.	Invalid packet received on VC in state P2 (DTE CALL REQUEST).
22 (CLEAR)	Packet type invalid for state P3.	Invalid packet received on VC in state P3 (DCE Incoming Call).
23 (CLEAR, RESET)	Packet type invalid for state P4.	Invalid packet received on VC in state P4 (Data Transfer).
24 (CLEAR)	Packet type invalid for state P5.	Invalid packet received on VC in state P5 (Call Collision).

Table C-2 Packet Type Invalid (Continued)

25 (CLEAR)	Packet type invalid for state P6.	Invalid packet received on VC in state P6 (DTE Clear Request).
27 (RESET)	Packet type invalid for state D1.	Invalid packet received on VC in state D1 (Flow Control Ready).
29 (RESET)	Packet type invalid for state D3	Invalid packet received on VC in state D3 (DCE Reset Indication).

Table C-3 Packet Not Allowed

Diagnostic	ISO Description	Explanation
33 (RESTART, CLEAR, RESET)	Unidentifiable packet.	Packet that cannot be identified (3rd byte of the level 2 information field not defined in *CCITT X.25 Recommendation* received). REJECT or REGISTRATION packet received.
34 (CLEAR)	Call on one way logical channel.	If configured as DTE, CALL INDICATION received with logical channel identifier (LCI) corresponding to one-way logical channel outgoing. If configured as DCE (DXE), CALL INDICATION received with logical channel identifier (LCI) corresponding to one way logical channel incoming.
38 (RESTART, CLEAR, RESET)	Packet too short.	Received packet is less than minimum length. It may be a CALL INDICATION packet without address length or facility length fields, CLEAR or RESET INDICATION without cause field, or the packet length may be insufficient to hold address or facility fields.

Table C-3 Packet Not Allowed (Continued)

39 (RESTART, CLEAR, RESET)	Packet too long.	Received packet exceeds maximum length. The packet may be larger than allowed by the configuration and negotiations (i.e., fast select), or contain a field (i.e., call/clear user data) that is larger than the allowed size.
41 (RESTART, CLEAR, RESET)	RESTART or REGISTRA-TION packet with non-zero LCI.	RESTART REQUEST sent because RESTART INDICATION arrived with logical channel identifier other than 0.
42 (CLEAR)	Packet type not compatible with facility.	CALL ACCEPTED or CONFIRMATION packet received when CALL INDICATION or REQUEST had fast select with restriction on response facility.
43 (RESET)	Unauthorized interrupt confirmation.	INTERRUPT CONFIRM packet received when no matching INTERRUPT had been sent.
44 (RESET)	Unauthorized interrupt.	Sequence of two INTERRUPT INDICATIONs received and system sent no INTERRUPT CONFIRMATION packet.

Table C-4 Timer Expired

Diagnostic	ISO Description	Explanation
48 (CLEAR)	Timer expired (generic).	IP circuit is disconnected because it was inactive for more than the configured minimum time limit and the system must free the circuit to complete CALL REQUEST.
49 (CLEAR)	Timer expired for INCOMING CALL or DTE timer expired for CALL REQUEST.	CALL CONFIRM not received within the T21 time limit (set at approximately 200 seconds) after transmission of CALL REQUEST.

Table C-4 Timer Expired (Continued)

50 (CLEAR)	Timer expired for CLEAR INDICATION (or DTE timer expired or retransmission count surpassed for CLEAR REQUEST).	CLEAR CONFIRM not received within T23 time limit (set at approximately 180 seconds) after transmission of CLEAR REQUEST. After a certain number of CLEAR retries, system places logical channel in state P1 (Ready).
51 (CLEAR, RESET)	Timer expired for RESET INDICATION (or DTE timer expired or retransmission count surpassed for RESET REQUEST).	RESET CONFIRM not received within T22 time limit, set at approximately 180 seconds, after transmission of RESET REQUEST. After a certain number of RESET retries, the system places logical channel in state D1 (Flow Control Ready) if PVC, or in state P6 (DTE Request) by sending CLEAR REQUEST if SVC. Count of RESET retries has been surpassed. CLEAR REQUEST has been sent.
52 (RESTART)	Timer expired for RESTART INDICATION (or DTE timer expired or retransmission count surpassed for RESTART REQUEST).	RESTART CONFIRM not received within T20 time limit (set at approximately 180 seconds) after transmission of RESTART REQUEST. SABM frame received while in state R2 (DTE RESTART REQUEST). RESTART REQUEST is transmitted.

Table C-5 Call Setup, Call Clearing, or Registration Problem

Diagnostic	ISO Description	Explanation

Table C-5 Call Setup, Call Clearing, or Registration Problem (Continued)

65 (CLEAR)	Facility/ registration code not allowed.	Packet size negotiation, Window Size Negotiation or Throughput Class Negotiation facility codes found in the facility field of the call setup packet when corresponding negotiation had not been configured. In CALL REQUEST, condition results in error returned to user instead of generation of CLEAR. If facility checking was configured, a facility code not allowed by configured level (1980 or 1984) was found **before** the first facility marker. In CALL REQUEST, this condition results in an error returned to user instead of generation of CLEAR. If facility checking was configured along with rejection of duplicate facilities, a combination of invalid facility codes was detected. The combination may indicate that the basic and extended format of given facility are present or that both the Closed User Group and Closed User Group with Outgoing Access facilities are present. In CALL REQUEST, this condition results in an error instead of a CLEAR. Facility marker 00 3B (hex) found, which corresponds to “internal” facilities. This facility marker is not allowed by either application or network. In CALL REQUEST, this condition results in an error returned to the user instead of CLEAR.

Table C-5 Call Setup, Call Clearing, or Registration Problem (Continued)

66 (CLEAR)	Facility parameter not allowed.	Parameter field of a packet size negotiation, window size negotiation or throughput class negotiation facility in a call setup packet has been found to be invalid because it contains values not allowed by the configured level (1980 or 1984). Parameter field of a packet size negotiation, window size negotiation, or throughput class negotiation facility in a CALL ACCEPTED or CALL CONFIRM packet has been found to be invalid because it contains values outside of the allowed range specified by the configuration and previous call setup packet. For DDN configured interfaces, the parameter field of the DDN standard or DDN precedence facility in a call setup packet has been found to be invalid because it contains values not allowed by the *DDN X.25 Host Interface Specification (BBN83)*.
67 (CLEAR)	Invalid called address.	Unknown called address. Called address contains a non-BCD digit (i.e., a half byte with a value other than 0 through 9).
68 (CLEAR)	Invalid calling address.	Calling address contains a non-BCD digit (i.e., a half byte with a value other than 0 through 9). Last digit of address field is not 0 and the total number of digits present in address field is odd.

Table C-5 Call Setup, Call Clearing, or Registration Problem (Continued)

69 (CLEAR)	Invalid facility/ registration length.	Length of the facility field is greater than the maximum allowed by the configured revision level of *CCITT X.25 Recommendation* (63 for 1980, 109 for 1984). The condition will result in an error returned to the user instead of generation of CLEAR if it is detected in CALL REQUEST. No combination of facilities can equal the value indicated in the packet's facility length field. If this condition is detected in the CALL REQUEST packet it produces an error instead of a CLEAR. System could not add negotiation facilities without the facility field becoming larger than allowed by the configured revision level of the *CCITT X.25 Recommendation* (63 for 1980, 109 for 1984). This can happen only if the configured negotiation facility codes are not present in the facility field.
70 (CLEAR)	Incoming call barred.	Configuration and status of interface disallows the opening of new inbound connection.
71 (CLEAR)	No logical channel available.	No VC suitable for placing a connection in READY (P1) state. Generated when CALL REQUEST cannot be completed successfully. ENOSPC was returned to the caller if condition was detected in synchronous manner. The number of internal connections prohibits new connections being allocated for CALL INDICATION.
72 (CLEAR)	Call collision.	When configured as DCE, CALL INDICATION was received on logical channel where CALL REQUEST had been sent. CALL REQUEST was cleared because the CALL INDICATION has priority.

Table C-5 Call Setup, Call Clearing, or Registration Problem (Continued)

73 (CLEAR)	Duplicate facility requested.	Same facility code appeared twice in the facility field. Error is returned to caller if condition was detected in CALL REQUEST. CLEAR was not generated.
76 (CLEAR)	Facility not provided when expected.	When system was configured to handle IP over X.25 as per DDN specifications, CALL INDICATION for IP did not have DDN standard selection facility code.

Table C-6 CATEGORY - Miscellaneous

Diagnostic	ISO Description	Explanation
83 (CLEAR)	Inconsistent Q bit setting.	The connection was cleared due to an inconsistent Q bit found in the packet.

Table C-7 International Problem and Maintenance

Diagnostic	ISO Description	Explanation
113 (CLEAR)	Remote network problem.	The remote network is not operational.
115 (CLEAR)	International link out of order.	Connection to the PSN is not currently available (network not operational).
121 (CLEAR)	Unknown called DNIC.	The CALL could not be routed because the DNIC in the called address is unknown.
122 (CLEAR, RESET)	Maintenance action.	System administrator shut down specific VC or the entire interface.

Table C-8 DTE-Specific Signals

Diagnostic	ISO Description	Explanation

Table C-8 DTE-Specific Signals (Continued)

160 (CLEAR, RESET)	DTE-specific signal (generic).	CALL INDICATION with reverse charge requested has been received, and reverse charge is configured to be rejected by system. CALL INDICATION with reverse charge requested has been received by IP and reverse charge is configured to be rejected for that IP/X.121 address pair. Check is done via IP to X.121 address map table. CALL INDICATION with reverse charge requested has been received by IP and no entry in the IP-to-X.121 address map table exists for this X.121 address. Internal problem which may be generated by components of X.25/9000 architecture.
161 (RESTART)	DTE operational.	Level 2 is coming up or a SABM frame is received. System sends RESTART REQUEST indicating host is up.
162 (CLEAR, RESET)	DTE not operational.	Either level 3 or level 2 detected as down. Request delivered when network interface was not operational. Condition usually results in an error returned to the application instead of CLEAR. Diagnostic is used when RESTART INDICATION has been received by interface. Interface is going down. RESTART REQUEST was sent on network.
163 (CLEAR, RESET)	DTE resource constraint.	Lack of memory to establish, reset, or clear a call, to process data, expedite data or acknowledgments, or to perform internal operation. This may indicate lack of network memory, or resource constraints on card. When resource constraint appears on connected circuit, CLEAR will be sent only if no recovery is possible.

Table C-8 DTE-Specific Signals (Continued)

164 (CLEAR)	Fast select not subscribed.	CALL INDICATION requiring fast select facility has been received and fast select is not configured. Results in error when detected on CALL REQUEST.
225 (CLEAR)	Disconnection (transient condition).	X.25 subsystem is down when X.25/9000 to IP subsystem is servicing CALL CONFIRMATION or incoming DATA.

Table C-9 OSI Network Service Problem

Diagnostic	ISO Description	Explanation
227 (CLEAR)	Connection rejection. Cause unspecified (transient condition).	Due to internal constraints at high levels other than lack of network memory, CALL INDICATION cannot be serviced.
224 (CLEAR, RESET)	OSI Network Service problem.	OSI network service not operational.
228 (CLEAR)	Connection rejection. Reason unspecified. (Permanent condition) (Transient condition).	X.25 to IP subsystem clears incoming CALL INDICATION which has calling address not present in IP to X.121 mapping table. If call were accepted, circuit would be half-duplex enabling remote node to send and receive while local host would only be able to receive. When X.25/IP output routine has data addressed to remote node it must be able to map the IP address to the X.121 address. If it cannot do so, the IP packet is discarded. Clearing the INCOMING CALL when the remote's X.121 address is not in the map table prevents a half-duplex circuit from being established.

Table C-9 OSI Network Service Problem (Continued)

231 (CLEAR)	Connection rejection. NSAP unreachable (transient condition).	No listen socket capable of servicing CALL INDICATION. X.25 subsystem is down when X.25-to-IP subsystem is servicing CALL INDICATION.
232 (CLEAR)	Connection rejection. NSAP unreachable (permanent condition).	CALL INDICATION with first byte of call user data other than CC (hex) received on system which does not have X.25/9000 programmatic access configured in kernel.
233 (RESET)	Reset. Cause unspecified.	Message (sequence of packets with the M bit set followed by packet with M bit not set) received, and size exceeds maximum size allowed by application for inbound data message. Corresponds to specific out-of-band event delivered to user. Data was received on VC without a user, that is, an IP or programmatic access application. X.25 subsystem was down when X.25 to IP subsystem serviced incoming DATA on permanent VC.
234 (RESET)	Reset. Congestion.	Error returned to the application due to network congestion condition.
235 (CLEAR)	NSAP address unknown.	CALL INDICATION was received with unknown NSAP address (first byte of call user data).
251 (RESET)	Reset. Cause unspecified.	System failed to synchronize data transfer to interface.

Table C-10 Higher Level Initiated

Diagnostic	ISO Description	Explanation

Table C-10 Higher Level Initiated (Continued)

241 (CLEAR)	Disconnection normal.	Application closed or shut down socket and did not specify cause/diagnostic combination to be used on socket. This is the default. If available, application-specified cause/diagnostic combination is used. This is not an error condition.
242 (CLEAR)	Disconnection abnormal.	Application using a socket supporting virtual connection crashed. Diagnostic, along with a cause code of 0, is used regardless of setting of cause/diagnostic combination. Related to `x29server` process.
243 (CLEAR)	Disconnection. Incompatible info in user data.	Application closed or shut down socket because of incompatible information in the user data field. Related to the `x29server` process.
244 (CLEAR)	Connection rejection. Reason unspecified (transient condition).	Application rejected the CALL for an unspecified reason. No listen socket was found. Related to the `x29server` process.
245 (CLEAR)	Connection rejection. Reason unspecified (permanent condition).	Application rejected the CALL for an unspecified reason. No listen socket was found. Related to the `x29server` process.
248 (CLEAR)	Connection rejection. Incompatible info in user data.	Application closed or shutdown socket because of a problem with the user data information. Related to the `x29server` process.

Table C-10 Higher Level Initiated (Continued)

250 (RESET)	Reset. User re-synchronizatio n.	PVC is claimed by programmatic access socket. RESET is sent as part of socket initialization procedure. Code is used if the application did not specify which cause/diagnostic combination to use. Application requires RESET to be sent. No cause/diagnostic combination is specified on socket.